#16
AF & 10f3

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

APPLICANT(s):	Williams et al.		
SERIAL NO.:	09/507,945	ART UNIT:	2174
FILING DATE:	02/22/2000	EXAMINER:	Muhebbullah, Sajeda
TITLE:	A COMMUNICATION TERMINAL HAVING A PREDICTIVE EDITOR APPLICATION		
ATTORNEY DOCKET NO.:	684-009255-US (PAR)		

Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450
ATTENTION: BOARD OF PATENT APPEALS AND INTERFERENCES

APPELLANTS' BRIEF
(37 C.F.R. §1.192)

Sir:

This is an appeal from the final rejection of the claims in the above-identified application. A Notice of Appeal was mailed on April 23, 2003. The fees required under 37 C.F.R. §1.17 are being submitted herewith. This brief is being submitted in triplicate. The appendix of claims are attached hereto.

I. REAL PARTY IN INTEREST

The real party in interest in the subject application is the assignee, Nokia Corporation, Helsinki, Finland.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

III. STATUS OF CLAIMS

Claims 1-3 stand rejected under 35 U.S.C. §103(a), based on the cited reference, King, US Patent No. 6,011,554 in view of the teachings of Frederiksen, US Patent No. 6,185,295. Claims 4 and 12 stand rejected under 35 U.S.C. §103(a), based on the cited reference, King. Claims 5, 6, and 11 stand rejected under 35 U.S.C. §103(a) based on the cited references, King, in view of the teachings of the cited reference Schroeder, US Patent No. 5,797,098. Copies of the cited references are attached as Exhibit B-D. Claims 1-6, 11, and 12 are presented for consideration in this appeal and are contained in Exhibit A.

IV. STATUS OF AMENDMENTS

An amendment was filed after final rejection to overcome the objection of the Examiner to the wording of the Abstract. It is not clear from the record whether or not this amendment was entered. The proposed amendment is contained in the office action attached hereto as Exhibit E.

V. SUMMARY OF INVENTION

In the embodiment of the invention, as described in claims 1-3, a communications terminal 1 is provided with a predictive editor

which operates to predict a word from an ambiguous string of key strokes. Referring to figures 2-4 and the text of the specification, in particular the text contained on page 6, line 10 to page 8, line 22, the processing module 61 keeps a record of the key sequence entered. Predictive editor processor 62 refers to user vocabularies 41a-41n that store vocabulary obtained from a variety of sources including applications programs. Predictive editor processor 62 searches the available vocabularies for strings that at least partly match the entered key strokes and generates a list 50 containing possible candidates. The communications terminal 1 is also supplied with an independent application program, for example, phone book 44 which is served in normal operation by a dedicated memory. The control processor 61 of the terminal 1 is programmed to automatically store words, entered by the application program in the application memory, to the memory 17 used by the predictive editor in order to supplement the predictive editor vocabulary.

A second embodiment of this invention, is described in independent claim 4. In this embodiment a communication terminal 1, having predictive editor processor 62, is also provided with a second editor as described at page 21, line 17 to page 22, line 8 for using unambiguous key strokes to edit the results of the predictive editor.

VI. ISSUES PRESENTED FOR REVIEW

A. The first issue presented for review is the propriety of the Examiner's rejection of claims 1-3 under 35 U.S.C. §103(a) based on the cited reference, King, US Patent No. 6,011,554 ("King") in view of the teachings of Frederiksen, US Patent No. 6,185,295

("Frederiksen"). The rejection is contained in the Office Action mailed December 24, 2002. Copies of the cited references are attached as Exhibits B and C, respectively.

B. The second issue presented for review is the propriety of the Examiner's rejection of claims 4 and 12 under 35 U.S.C. §103(a) based solely on the cited reference, King. The rejection is contained in the Office Action mailed December 24, 2002.

C. The third issue presented for review is the propriety of the Examiner's rejection of claims 5, 6, and 11 under 35 U.S.C. §103(a) based on the cited reference, King, in view of the teachings of Schroeder, US Patent No. 5,797,098 ("Schroeder"). The rejection is contained in the Office Action mailed December 24, 2002. A copy of Schroeder is attached as Exhibit D.

D. A fourth issue is also presented for review with respect to the objections to the Abstract. Applicant timely presented an Amendment After Final which amended the Abstract to overcome this objection. This amendment is contained in Exhibit E.

VII. GROUPING OF CLAIMS

The claims under consideration are attached as Exhibit A. The claims are arranged in two distinct groups, namely, independent claim 1 and its dependent claims 2 and 3, and independent claim 4 and its dependent claims 5,6, and 11. The Examiner has differentiated these groups of claims in his rejections.

The groups of claims do not stand or fall together, and the patentability of the groups of claims will be separately argued below.

VIII. ARGUMENT

A. With respect to issue A relating to claims 1-3, the cited art fails to teach a system that utilizes a predictive editor having a dedicated memory, where said predictive editor is used in conjunction with applications software also having a dedicated memory and further where the predictive editor and application software is controlled by a processor which automatically searches and copies words from the memory of the application software into the predictive memory. .

The Examiner relies primarily on the disclosure of King. King describes a compact keyboard which provides a reduced number of keys. Each key has multiple letters, which may be entered by depression of the key. A predictive editor is used to facilitate interpretation of the key strokes by reference to a predetermined dictionary which is generated based on frequency of use. This is described concisely in the Abstract of King.

The Examiner admits that King does not disclose the automatic entry of usable words in its predictive editor dictionary from other application software as required by claims 1-3 of Applicants' invention.

In order to remedy this deficiency in King, the Examiner cites the reference Frederiksen. Frederiksen discloses a speed dial system which has a limited memory directly accessible for

dialing of important numbers. A larger database of phone numbers is provided through the use of a SIM card. In the system of Frederiksen, the speed dialing status of a phone number in the larger memory can be exchanged for that of a number in a speed dial location. The data is not processed in the second memory, but rather is only waiting to be transferred.

In the Office Action of December 24, 2002, the Examiner indicates that the memory on the SIM card can be copied to the memory of the phone. This is an exaggerated characterization, as only a few of the numbers on the SIM card may be exchanged with numbers in the phone's speed dial memory. The system of Frederiksen facilitates identifying a particular number as a speed dial candidate, nothing more. (Please refer to column 1, Lines 41-57 of Frederiksen.) There is no mention of a predictive editor having a dictionary of words which can be supplemented by use of independent application memory. In the operation of the subject invention, the dictionary of the predictive editor is automatically supplemented with words from the independent application memory by the main processing unit of the mobile terminal.

In combining the teachings of the references King and Frederiksen, the Examiner proposes to combine the teachings of a reference describing a compact keyboard for a personal digital assistant (King) with a reference teaching a speed dial facilitator for a mobile telephone (Frederiksen). The Examiner gives no indication as to what incentive is present in the references to encourage a person skilled in the art to make the combination. Neither King nor Frederiksen, either alone or in combination, disclose or suggest a concept for searching for

matches in text input from other applications than the application in which the predictive editor is employed. There is nothing in these references that would lead to such a combination. It is submitted that the Examiner has failed to support a *prima facie* case of obviousness.

According to basic tenets of patent law in order to support an obviousness rejection, there must be some suggestion of the desirability of making the modification, aside from the subject application. The claimed invention must be considered as a whole and the references must suggest the desirability and thus the obviousness of making the modification. The references must also be viewed without the benefit of hindsight. (See MPEP sections 706.02(a) and 2141). Applicant submits that the modification of the teachings of King and Frederiksen as proposed by the Examiner in order to obtain the invention, as described in the claims 1-3 of this application, would not have been obvious to one skilled in the art. There is no indication that such a modification would be desirable.

For further argument, if the proposed combination is assumed, the combined system would still not include a predictive editor having a dedicated memory, where said predictive editor is used in conjunction with applicants software also having a dedicated memory and further where the predictive editor and application software is controlled by a processor which automatically searches and copies words from the memory of the application software into the predictive editor memory.

Thus, claims 1-3 are not unpatentable over King in view of Frederiksen under 35 U.S.C. §103(c).

B. With respect to issue B relating to the rejection of claims 4 and 12 the cited references fail to disclose a predictive editor that is used in conjunction with a second editor application, the second editor usable for the purpose of amending the words obtained by the predictive editor.

In stating his position, in the Office Action of December 24, 2002, with respect the rejection of claim 4, the Examiner states:

> "King teaches an editor application for entering words in an unambiguous form. However, King does not explicitly disclose the editor application to be used to revise, delete and/or combing words."

This is an understatement, as the editor function of King does nothing but select or reject the results of the predictive editor function. King does not disclose or suggest removing words from a non-transient dictionary as is claimed by Applicant. To remedy this deficiency, the Examiner indicates that official notice is given that such managing operations for user defined lists were well known. Yet this statement remains unsupported by any reference. Considering this statement further, it is clear that the Examiner has missed an important feature of claim 4, namely:

> "A second editor application controlled by said processor means for entering key strokes in an unambiguous form; wherein said second editor is used to revise, delete, and/or combine said matching words generated by said first editor application."

The Examiner's speculation with regard to the editing capabilities of the predictive editor of the subject application

does not remedy the further deficiency of King, namely, that there is no second editor application. Editing in the system of King is rudimentary at best and includes no second editing capability.

Claim 12 is dependent on claim 4 and features the use of the second editor application to remove or maintain words from or in the selection process. This is also not disclosed or suggested by King for the same reasons stated above. Furthermore, it is again submitted that there is no motivation or suggestion to modify King to achieve Applicant's invention according to claims 4 and 12. There is nothing at all in King that would motivate one to modify King to achieve a communication terminal in which a second editor application is used to revise, delete and/or combine matching words generated by a first editor application. Thus, claims 4 and 12 are not unpatentable over King.

C. With respect to issue C, relating to claims 5, 6, and 11, the cited references fail to disclose a predictive editor which is used in conjunction with a second editor application, where the second editor is usable for the purpose of amending the words obtained by the predictive editor.

Claims 5, 6 and 11 are each dependent on claim 4, accordingly the rejection must also fail according to the above stated arguments relative to issue B.

In the examination of the subject application, it does not appear that the Examiner has considered the claims as a whole, but has dismantled the claims and pursued a search for the individual features. It is well settled that "the actual

determination of the issue requires an evaluation in the light of the findings in those inquiries of the obviousness of the claimed invention as whole, not merely the differences between the claimed invention and the prior art." (Graham v. John Deer Co., 383 U.S. 17). The court admonishes in In re Fritch, 972 F.2d 1260 as follows:

> "It is impermissible to use the claimed invention as an instruction manual or "template" to piece together the teachings of the prior art so that the claimed invention is rendered obvious. This court has previously stated that "one cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention."

D. With respect to issue D, it is Applicant's position that these amendments should have been entered to place the claims in better condition for appeal. Applicant submits that such amendments fully overcome the Examiner's objections.

IX. SUMMARY

It is respectfully submitted that all of the claims, as presented, are clearly novel and patentable over the prior art of record. Accordingly, the Board of Appeals is respectfully requested to favorable consider the rejected claims and to reverse the final rejections, thereby enabling this application to issue as a U.S. Letters Patent.

A check in the amount of $430 is enclosed for a one-month extension of time and appeal brief fee. The Commissioner is hereby authorized to charge payment for any fees associated with this communication or credit any over payment to Deposit Account No. 16-1350.

Respectfully submitted,

Geza C. Ziegler, Jr.
Reg. No. 44,004

3 July 2003
Date

Perman & Green, LLP
425 Post Road
Fairfield, CT 06824
(203) 259-1800 Ext. 134
Customer No.: 2512

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service on the date indicated below as first class mail in an envelope addressed to the Commissioner of Patents, P.O. Box 1450, Alexandria, VA 22313-1450, Attention: BOARD OF PATENT APPEALS AND INTERFERENCES

Date: 7/3/03

Signature: D. Boland
Person Making Deposit

APPENDIX A

1. (previously amended) A communication terminal having:

a display;

a keypad for use in the operation of said communication terminal having a plurality of keys associated with several letters each;

processor means controlling the display in accordance with the operation of the keypad;

a predictive editor program for generating an output containing word matching a received string of ambiguous key strokes, said predictive editor program having a number of associated vocabularies including at least one language dependent dictionary and at least one dictionary receiving user defined inputs stored in a first memory which serves said predictive editor program;

an editor application controlled by the processor means communicates with said predictive editor programs for generating matching words based on an ambiguous string of key strokes;

at least one applications program independent of said predictive editor program:

second memory means of the communication terminal independent of said first memory means for storing user inputted data

in an electronic database, said second memory means serving said at least one applications program; and

wherein said processor means automatically searches said second memory means for words and copies these words into said at least one dictionary for receiving user defined inputs and associated with said predictive editor program.

2. (original) A communication terminal according to claim 1 wherein said second memory means is an electronic phonebook database containing names and associated phone numbers.

3. (original) A communication terminal according to claim 2 wherein said electronic phonebook database is stored on a Subscriber Identify Module in a cellular phone.

4. (previously amended) A communication terminal having:

a display;

a keypad having a plurality of keys associated with several letters each;

processor means controlling the display in accordance with the operation of the keypad;

a predictive editor program for generating an output containing words matching a received string of ambiguous key strokes, said predictive editor program having a number of associated vocabularies including at least one language

dependent dictionary and at least one dictionary receiving user defined inputs;

an editor application controlled by the processor means communications with said predictive editor programs for generating matching words based on an ambiguous string of key strokes;

a second editor application controlled by said processor means for entering key strokes in an unambiguous form; wherein said second editor is used to revise, delete, and/or combine said matching words generated by said first editor application.

5. (previously amended) A communication terminal according to claim 11 wherein the processor means updates the storing time every time the word is used by the editor application.

6. (original) A communication terminal according to claim 5 wherein the dictionary containing the unambiguously entered words is built up as a cyclic buffer, where the word having the oldest storing time is removed from the memory when a new word is added and the buffer is full.

7-10. (canceled)

11. (previously added) A communications terminal according to claim 4, wherein said editor application stores words that have to be entered in an unambiguous way in one of said least one dictionary receiving user defined inputs;

said processor means associates a storing time for the unambiguously entered words stored in said dictionary receiving user defined inputs; and

said processor means maintains the dictionary containing the unambiguously entered words dependent on the storing time.

12. (previously added) A communications terminal according to claim 4, further including means adapted to remove or maintain words entered using said second editor application.

EXHIBIT B

US006011554A

United States Patent [19]

King et al.

[11] Patent Number: 6,011,554

[45] Date of Patent: Jan. 4, 2000

[54] **REDUCED KEYBOARD DISAMBIGUATING SYSTEM**

[75] Inventors: **Martin T. King**, Vashon, Wash.; **Dale L. Grover**, Lansing, Mich.; **Clifford A. Kushler**; **Cheryl A. Grunbock**, both of Vashon, Wash.

[73] Assignee: **Tegic Communications, Inc.**, Seattle, Wash.

[21] Appl. No.: **08/686,955**

[22] Filed: **Jul. 26, 1996**

Related U.S. Application Data

[63] Continuation-in-part of application No. 08/507,756, Jul. 26, 1995, Pat. No. 5,818,437

[60] Provisional application No. 60/021,180, Jun. 10, 1996.

[51] **Int. Cl.**[7] **G06F 15/00**

[52] **U.S. Cl.** **345/352**; 345/168; 704/10; 707/533

[58] **Field of Search** 345/326, 352, 345/353, 354, 168, 169; 704/3, 7, 9, 10; 707/102, 530, 533, 534

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,647,973	3/1972	James et al.	340/365
3,967,273	6/1976	Knowlton .	

(List continued on next page.)

FOREIGN PATENT DOCUMENTS

0 213 022	3/1987	European Pat. Off. .
0 319 193 A2	6/1989	European Pat. Off. .
0 464 726 A2	1/1992	European Pat. Off. .
0 540 147 A2	5/1993	European Pat. Off. .
0 651 315 A1	5/1995	European Pat. Off. .
0 689 122 A1	12/1995	European Pat. Off. .
0 732 646	9/1996	European Pat. Off. .
2 298 166	8/1996	United Kingdom .
82/00442	2/1982	WIPO .
PCT/US89/05745	6/1990	WIPO .
WO90/07149	6/1990	WIPO .
96/27947	9/1996	WIPO .
WO 97/04580	2/1997	WIPO .

OTHER PUBLICATIONS

Sugimoto, M. et al., "SHK: Single Hand Key Card for Mobile Devices," CHI 1996 (Apr. 13–18, 1996).

(List continued on next page.)

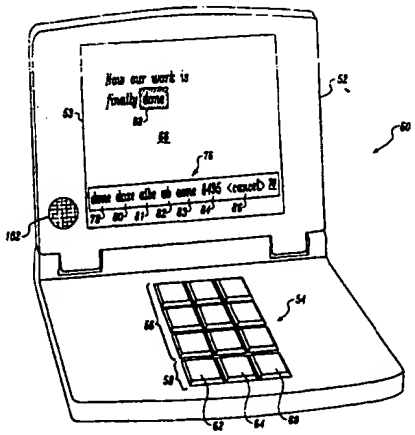
Primary Examiner—Ba Huynh

Attorney, Agent, or Firm—Christensen O'Connor Johnson & Kindness PLLC

[57] **ABSTRACT**

A reduced keyboard disambiguating system (**50**) having a keyboard (**54**) with a reduced number of keys. A plurality of letters and symbols are assigned to a set of data keys (**56**) so that keystrokes entered by a user are ambiguous. Due to the ambiguity in each keystroke, an entered keystroke sequence could match a number of words with the same number of letters. The disambiguating system includes a memory (**104**) having a number of vocabulary modules (**110**). The vocabulary modules contain a library of objects that are each associated with a keystroke sequence. Each object is also associated with a frequency of use. Objects within the vocabulary modules that match the entered keystroke sequence are identified by the disambiguating system. Objects associated with a keystroke sequence that match the entered keystroke sequence are displayed to the user in a selection list (**76**). The objects are listed in the selection list according to their frequency of use. An unambiguous select key (**60**) is pressed by a user to delimit the end of a keystroke sequence. The first entry in the selection list is automatically selected by the disambiguating system as the default interpretation of the ambiguous keystroke sequence. The user accepts the selected interpretation by starting to enter another ambiguous keystroke sequence. Alternatively, the user may press the select key a number of times to select other entries in the selection list. For words that are not in the vocabulary modules, a two-stroke or multiple-stroke method are used to unambiguously specify each letter. The disambiguating system simultaneously interprets all entered keystroke sequences as a word, as a two-stroke sequence, and as a multiple-stroke sequence. The multiple interpretations are automatically and simultaneously provided to the user in the selection list.

49 Claims, 27 Drawing Sheets



U.S. PATENT DOCUMENTS

4,191,854	3/1980	Coles .	
4,360,892	11/1982	Endfield .	
4,381,502	4/1983	Prame	340/365
4,426,555	1/1984	Underkoffler .	
4,427,848	1/1984	Tsakanikas .	
4,442,506	4/1984	Endfield .	
4,481,508	11/1984	Kamei et al. .	
4,549,279	10/1985	Lapeyre .	
4,649,563	3/1987	Riskin	379/97
4,661,916	4/1987	Baker et al. .	
4,674,112	6/1987	Kondraske et al. .	
4,677,659	6/1987	Dargan .	
4,754,474	6/1988	Feinson .	
4,791,408	12/1988	Heusinkveld .	
4,817,129	3/1989	Riskin	379/88
4,823,294	4/1989	Rouhani .	
4,846,598	7/1989	Livits .	
4,849,732	7/1989	Dolenc .	
4,866,759	9/1989	Riskin	379/97
4,872,196	10/1989	Royer et al. .	
4,891,777	1/1990	Lapeyre .	
5,031,206	7/1991	Riskin .	
5,035,205	7/1991	Schiller et al.	119/168
5,063,376	11/1991	Chang .	
5,065,661	11/1991	Hacker .	
5,067,103	11/1991	Lapeyre .	
5,087,910	2/1992	Guyot-Sionnest .	
5,131,045	7/1992	Roth .	
5,156,475	10/1992	Zilberman .	
5,163,084	11/1992	Kim et al. .	
5,200,988	4/1993	Riskin .	
5,214,689	5/1993	O'Sullivan .	
5,218,538	6/1993	Zhang .	
5,229,936	7/1993	Decker et al. .	
5,255,310	10/1993	Kim et al. .	
5,258,748	11/1993	Jones .	
5,281,966	1/1994	Walsh .	
5,289,394	2/1994	Lapeyre .	
5,305,205	4/1994	Weber et al. .	
5,317,647	5/1994	Pagallo .	
5,339,358	8/1994	Danish et al. .	
5,388,061	2/1995	Hankes .	
5,392,338	2/1995	Danish et al. .	
5,664,896	9/1997	Blumberg .	
5,786,776	7/1998	Kisaichi et al. .	
5,847,697	12/1998	Sugimoto .	

OTHER PUBLICATIONS

"Speed Keyboard for Data Processor," IBM Technical Disclosure Bulletin, vol. 23, pp. 838–839, Jul., 1980. ©IBM Corp., 1993.

Smith, Sidney L. et al., "Alphabetic Data Entry Via the Touch–Tone Pad: A Comment," *Human Factors,* 13(2), pp. 189–190, 1971.

Witten, I.H., "Principles of Computer Speech," New York: Academic Press, (1982), pp. 246–253 1982.

Levine, S.H., "An Adaptive Approach to Optimal Keyboard Design for Nonvocal Communication," *IEEE,* 1985.

Minneman, S.L., "A Simplified Touch–Tone® Telecommunication Aid for Deaf and Hearing Impaired Individuals," RESNA 8th Annual Conference, Memphis Tennessee, 1985.

Swiffin, A.L. et al., "PAL: An Effort Efficient Portable Communication Aid and Keyboard Emulator," RESNA 8th Annual Conference, Memphis, Tennessee, 1985.

Levine, S.H. et al., "Adaptive Technique for Customized Interface Design With Application to Nonvocal Communication," RESNA 9th Annual Conference, Minneapolis, Minnesota, 1986.

Levine, S.H. et al., "Computer Disambiguation of Multi–Character Key Text Entry: An Adaptive Design Approach," *IEEE,* 1986.

Foulds, R. et al., "Lexical Prediction Techniques Applied to Reduce Motor Requirements for Augmentative Communication," RESNA 10th Annual Conference, San Jose, California, 1987.

Foulds, R.A. et al., "Statistical Disambiguation of Multi–Character Keys Applied to Reduce Motor Requirements for Augmentative and Alternative Communication," *AAC Augmentative and Alternative Communication* (1987).

Levine, S.H. et al., "Multi–Character Key Text Entry Using Computer Disambiguation," RESNA 10th Annual Conference, San Jose, California, 1987.

Swiffin, A.L. et al., "Adaptive and Predictive Techniques in a Communication Prosthesis," *AAC Augmentative and Alternative Communication* (1987).

Kamphuis, H. et al., "Katdas; A Small Number of Keys Direct Access System," RESNA 12th Annual Conference, New Orleans, Louisiana, 1989.

Kreifeldt, J.G. et al., "Reduced Keyboard Designs Using Disambiguation," *Proceedings of the Human Factors Society 33rd Annual Meeting*–1989.

Arnott, J.L. et al., "Probabilistic Character Disambiguation for Reduced Keyboards Using Small Text Samples," *AAC Augmentative and Alternative Communication,* vol. 8 (Sep. 1992).

Oommen, B.J. et al., "Correction to 'An Adaptive Learning Solution to the Keyboard Optimization Problem," *IEEE Transactions on Systems, Man, and Cybernetics,* 22:5 (Oct., 1992).

Matias, E. et al., "Half–QWERTY: Typing With One Hand Using Your Two–Handed Skills," *Conference Companion,* CHI '94, (Apr. 24–28, 1994).

King, M.T., "JustType™–Efficient Communication with Eight Keys," *Proceedings of the RESNA '95 Annual Conference,* Vancouver, BC, Canada, 1995.

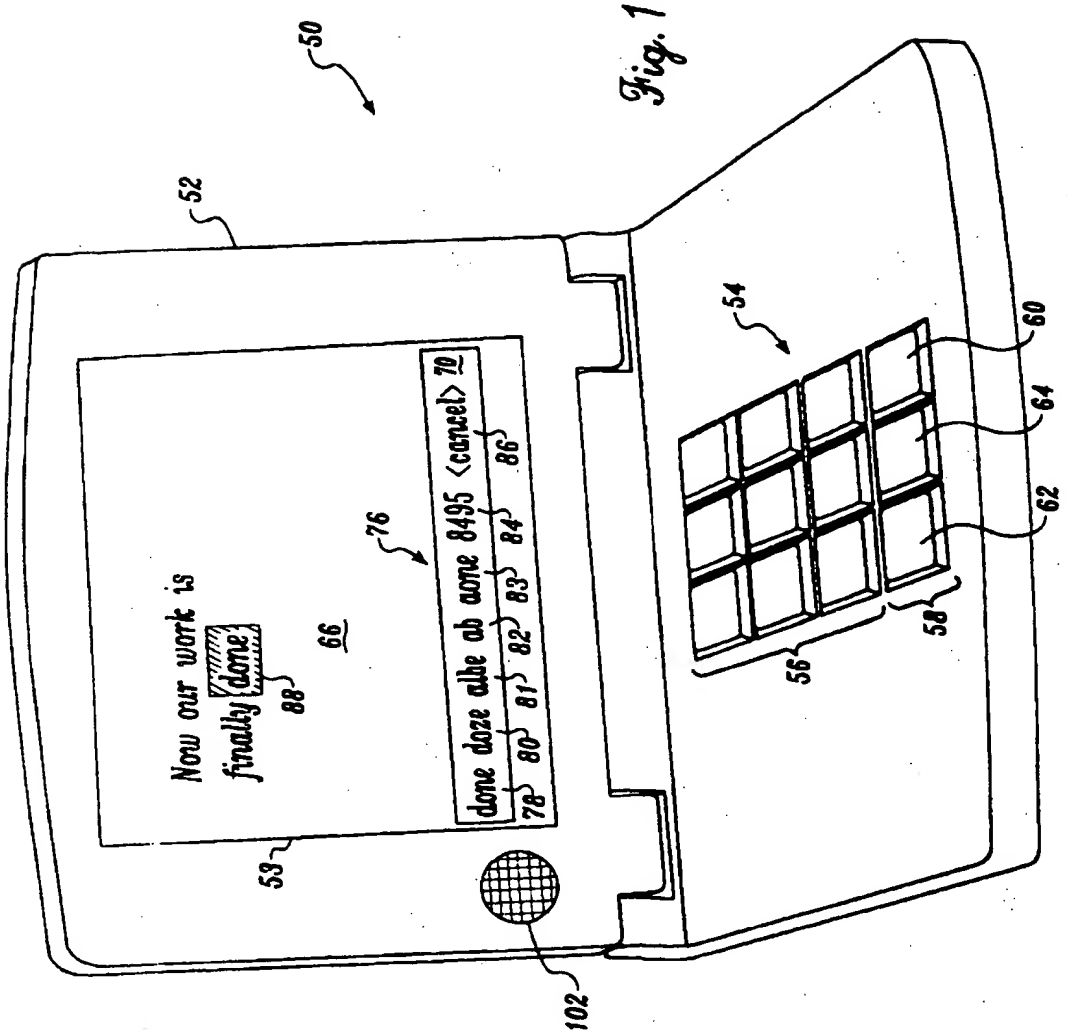

50
52
54
60
64
62
56
58
Now our work is
finally done
88
66
76
done doze albe ab aome 8495 <cancel> 70
78 80 81 82 83 84 86
53
102

Fig. 1A.

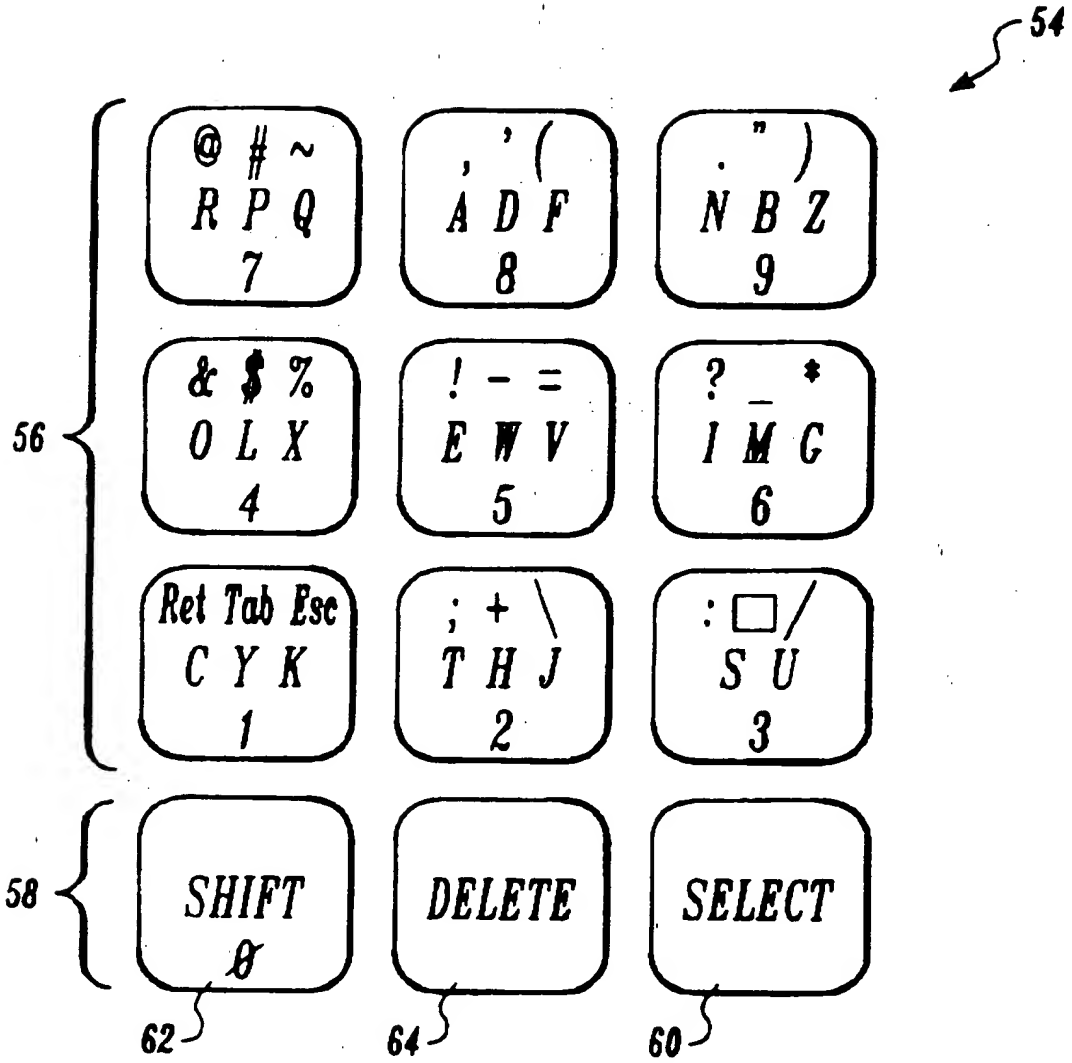

54
@ # ~
R P Q
7
, ' (
A D F
8
. ")
N B Z
9
& $ %
O L X
4
! - =
E W V
5
? _ *
I M G
6
56
Ret Tab Esc
C Y K
1
; + \
T H J
2
: □ /
S U
3
58
SHIFT
Ø
DELETE
SELECT
62
64
60

Fig. 1B.

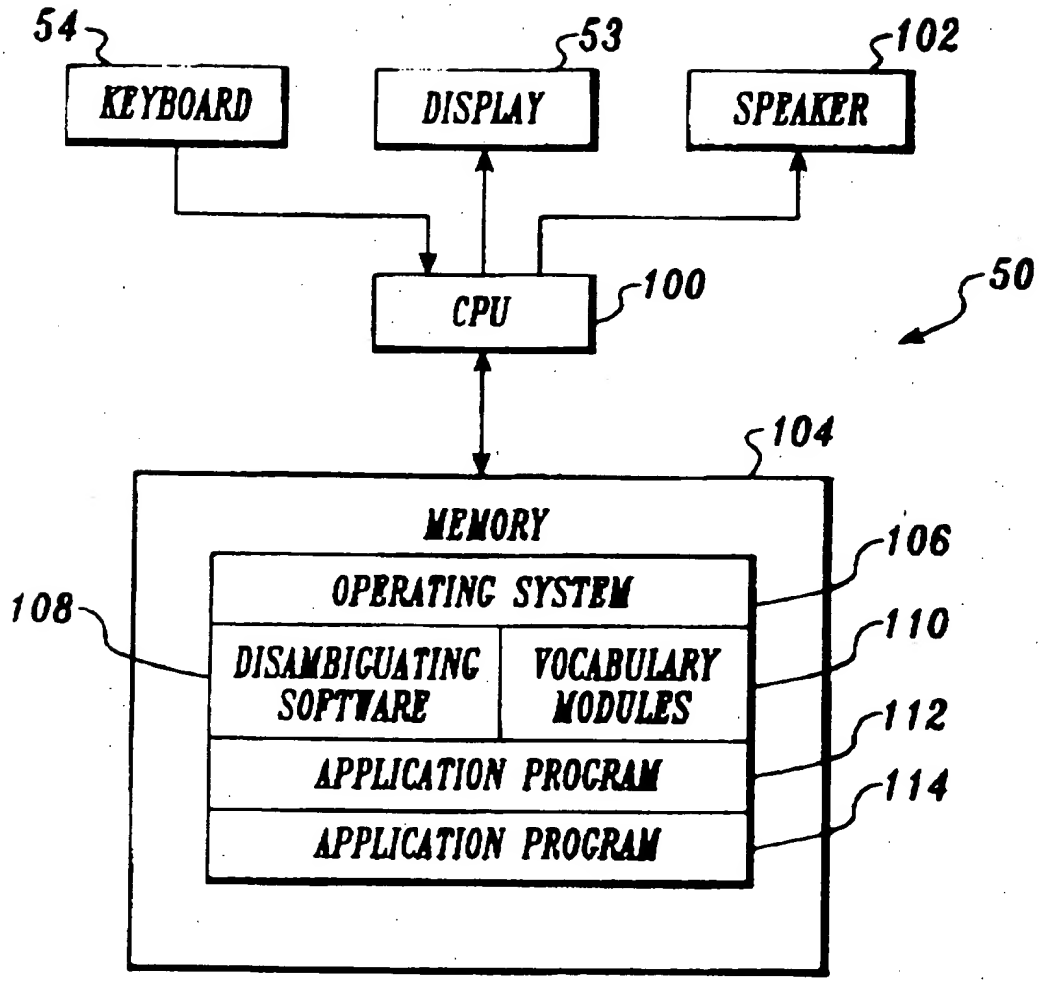

54
KEYBOARD
53
DISPLAY
102
SPEAKER
CPU
100
50
104
MEMORY
106
OPERATING SYSTEM
108
DISAMBIGUATING SOFTWARE
VOCABULARY MODULES
110
112
APPLICATION PROGRAM
114
APPLICATION PROGRAM

Fig. 2.

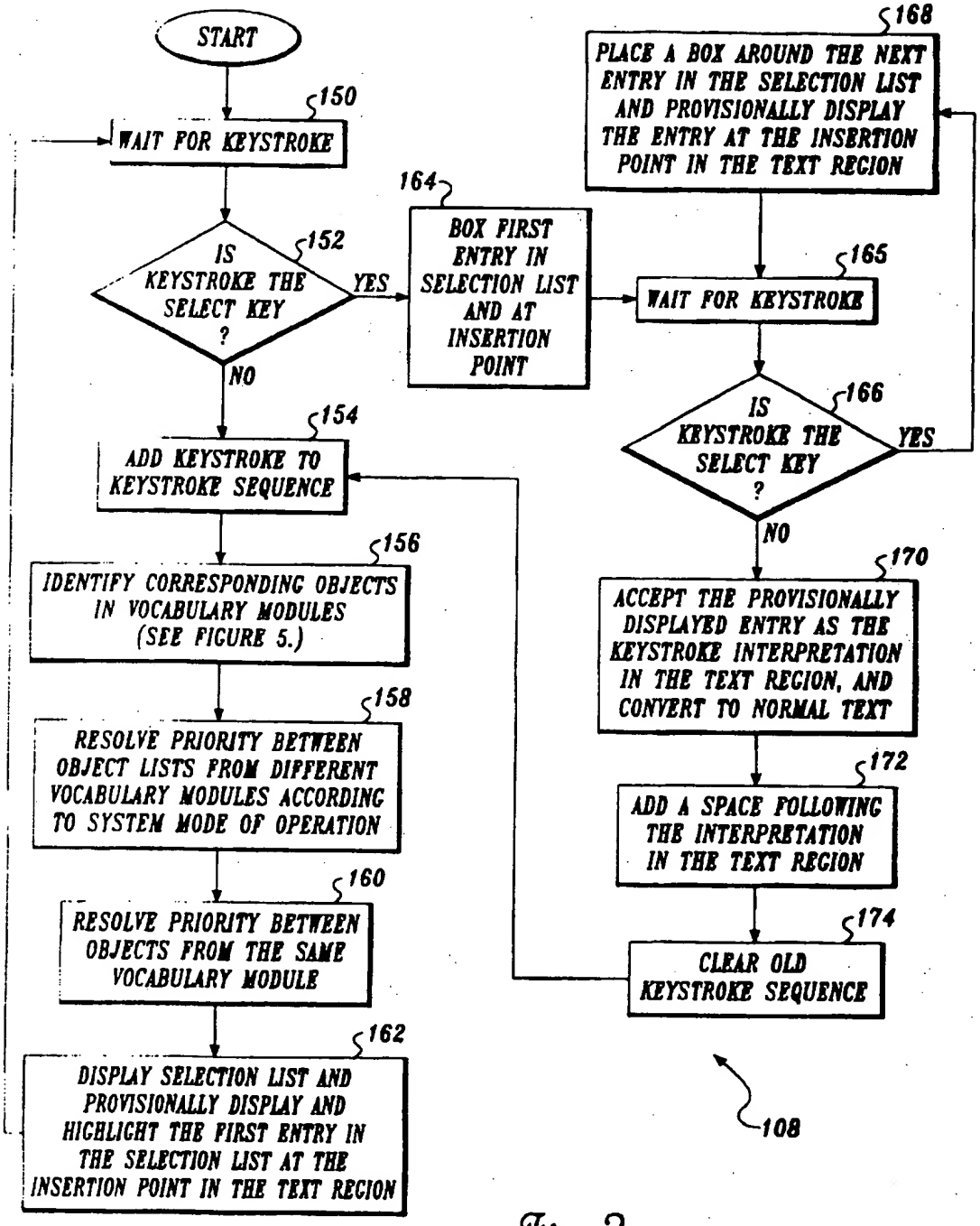

START
150
WAIT FOR KEYSTROKE
152
IS KEYSTROKE THE SELECT KEY ?
YES
NO
164
BOX FIRST ENTRY IN SELECTION LIST AND AT INSERTION POINT
165
WAIT FOR KEYSTROKE
166
IS KEYSTROKE THE SELECT KEY ?
YES
NO
168
PLACE A BOX AROUND THE NEXT ENTRY IN THE SELECTION LIST AND PROVISIONALLY DISPLAY THE ENTRY AT THE INSERTION POINT IN THE TEXT REGION
154
ADD KEYSTROKE TO KEYSTROKE SEQUENCE
156
IDENTIFY CORRESPONDING OBJECTS IN VOCABULARY MODULES (SEE FIGURE 5.)
158
RESOLVE PRIORITY BETWEEN OBJECT LISTS FROM DIFFERENT VOCABULARY MODULES ACCORDING TO SYSTEM MODE OF OPERATION
160
RESOLVE PRIORITY BETWEEN OBJECTS FROM THE SAME VOCABULARY MODULE
162
DISPLAY SELECTION LIST AND PROVISIONALLY DISPLAY AND HIGHLIGHT THE FIRST ENTRY IN THE SELECTION LIST AT THE INSERTION POINT IN THE TEXT REGION
170
ACCEPT THE PROVISIONALLY DISPLAYED ENTRY AS THE KEYSTROKE INTERPRETATION IN THE TEXT REGION, AND CONVERT TO NORMAL TEXT
172
ADD A SPACE FOLLOWING THE INTERPRETATION IN THE TEXT REGION
174
CLEAR OLD KEYSTROKE SEQUENCE
108

Fig. 3.

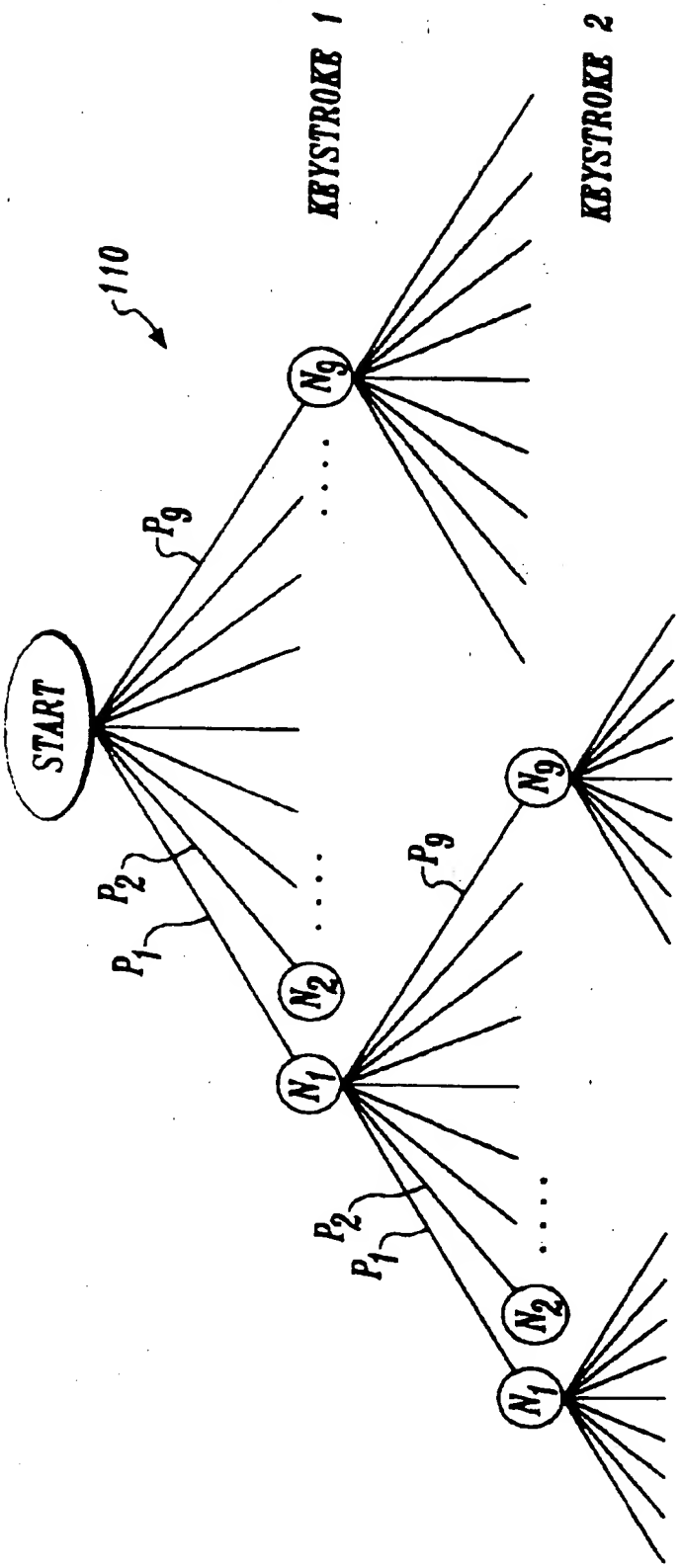


Fig. 4A.

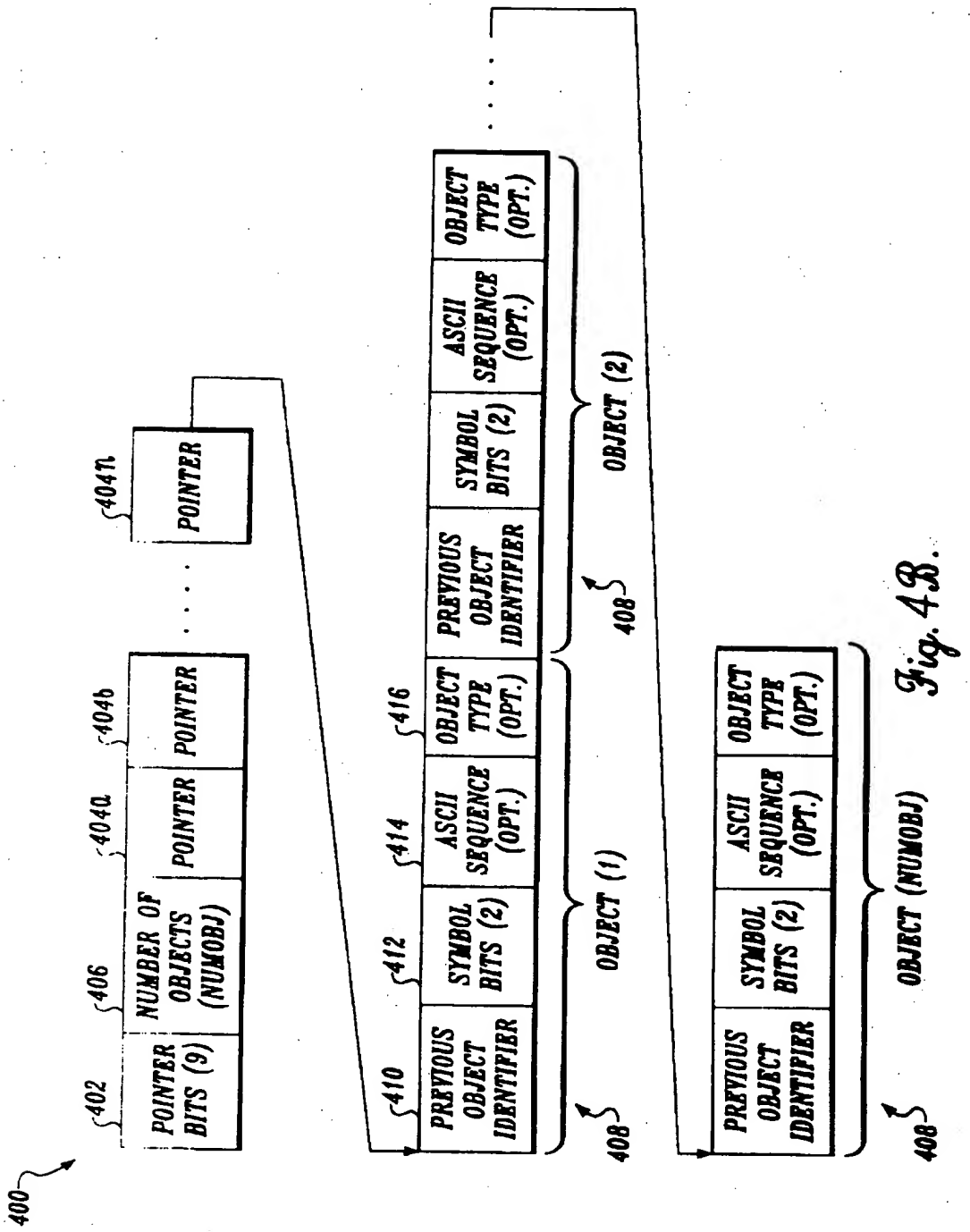

400
402
POINTER BITS (9)
406
NUMBER OF OBJECTS (NUMOBJ)
404a
POINTER
404b
POINTER
404n
POINTER
410
PREVIOUS OBJECT IDENTIFIER
412
SYMBOL BITS (2)
414
ASCII SEQUENCE (OPT.)
416
OBJECT TYPE (OPT.)
OBJECT (1)
408
OBJECT (2)
OBJECT (NUMOBJ)

Fig. 4B.

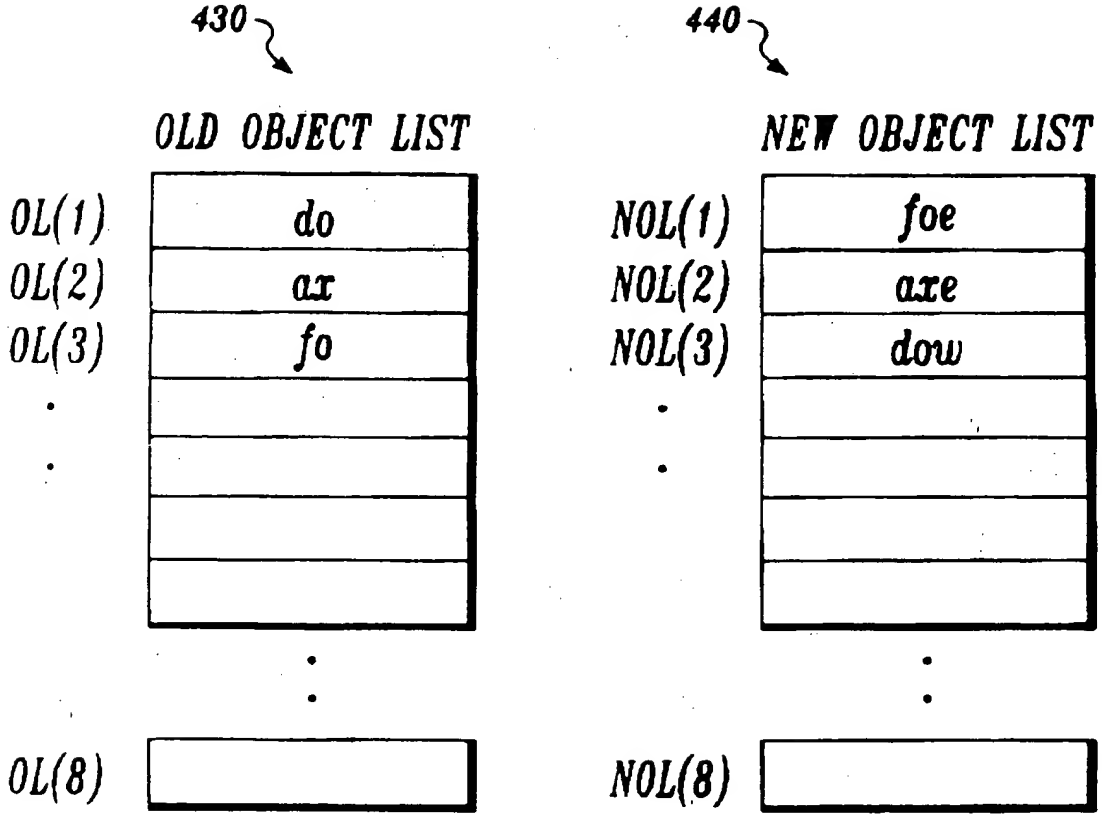

430
440
OLD OBJECT LIST
NEW OBJECT LIST
OL(1)
do
OL(2)
ax
OL(3)
fo
OL(8)
NOL(1)
foe
NOL(2)
axe
NOL(3)
dow
NOL(8)

Fig. 4C.

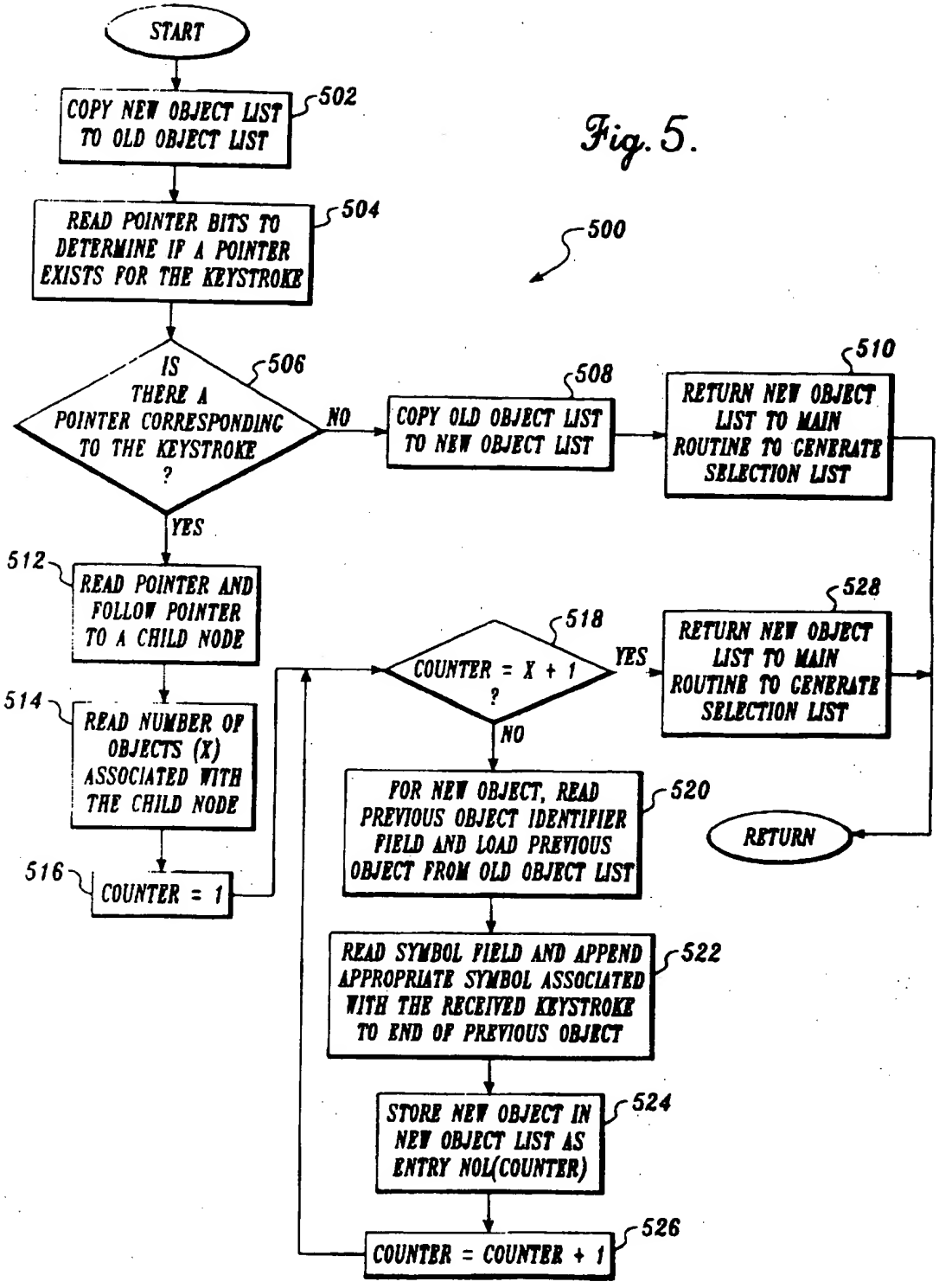

START
COPY NEW OBJECT LIST TO OLD OBJECT LIST
502
READ POINTER BITS TO DETERMINE IF A POINTER EXISTS FOR THE KEYSTROKE
504
500
IS THERE A POINTER CORRESPONDING TO THE KEYSTROKE ?
506
NO
COPY OLD OBJECT LIST TO NEW OBJECT LIST
508
RETURN NEW OBJECT LIST TO MAIN ROUTINE TO GENERATE SELECTION LIST
510
YES
READ POINTER AND FOLLOW POINTER TO A CHILD NODE
512
READ NUMBER OF OBJECTS (X) ASSOCIATED WITH THE CHILD NODE
514
COUNTER = 1
516
COUNTER = X + 1 ?
518
YES
RETURN NEW OBJECT LIST TO MAIN ROUTINE TO GENERATE SELECTION LIST
528
NO
FOR NEW OBJECT, READ PREVIOUS OBJECT IDENTIFIER FIELD AND LOAD PREVIOUS OBJECT FROM OLD OBJECT LIST
520
RETURN
READ SYMBOL FIELD AND APPEND APPROPRIATE SYMBOL ASSOCIATED WITH THE RECEIVED KEYSTROKE TO END OF PREVIOUS OBJECT
522
STORE NEW OBJECT IN NEW OBJECT LIST AS ENTRY NOL(COUNTER)
524
COUNTER = COUNTER + 1
526

Fig. 5.

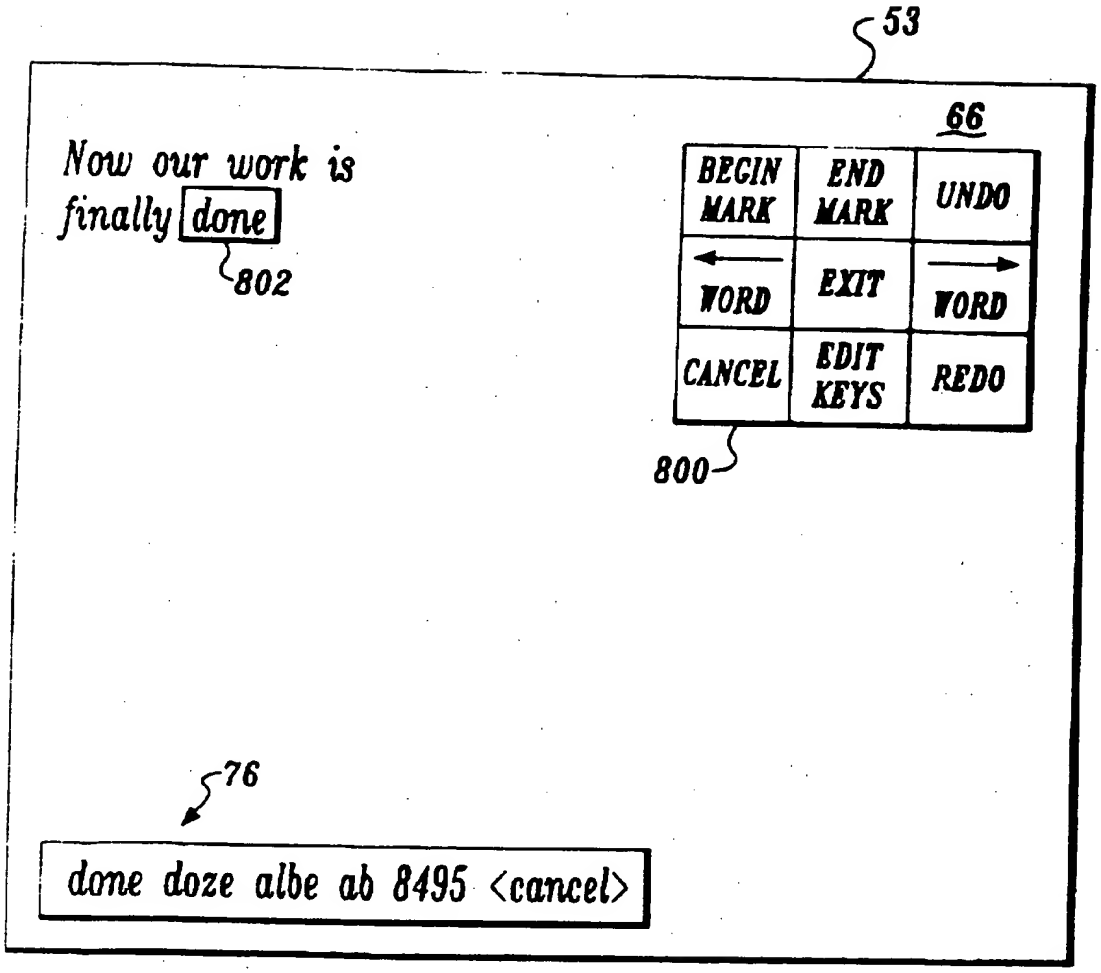

53
66
Now our work is finally done
802
BEGIN MARK
END MARK
UNDO
WORD
EXIT
WORD
CANCEL
EDIT KEYS
REDO
800
76
done doze albe ab 8495 <cancel>

Fig. 6.

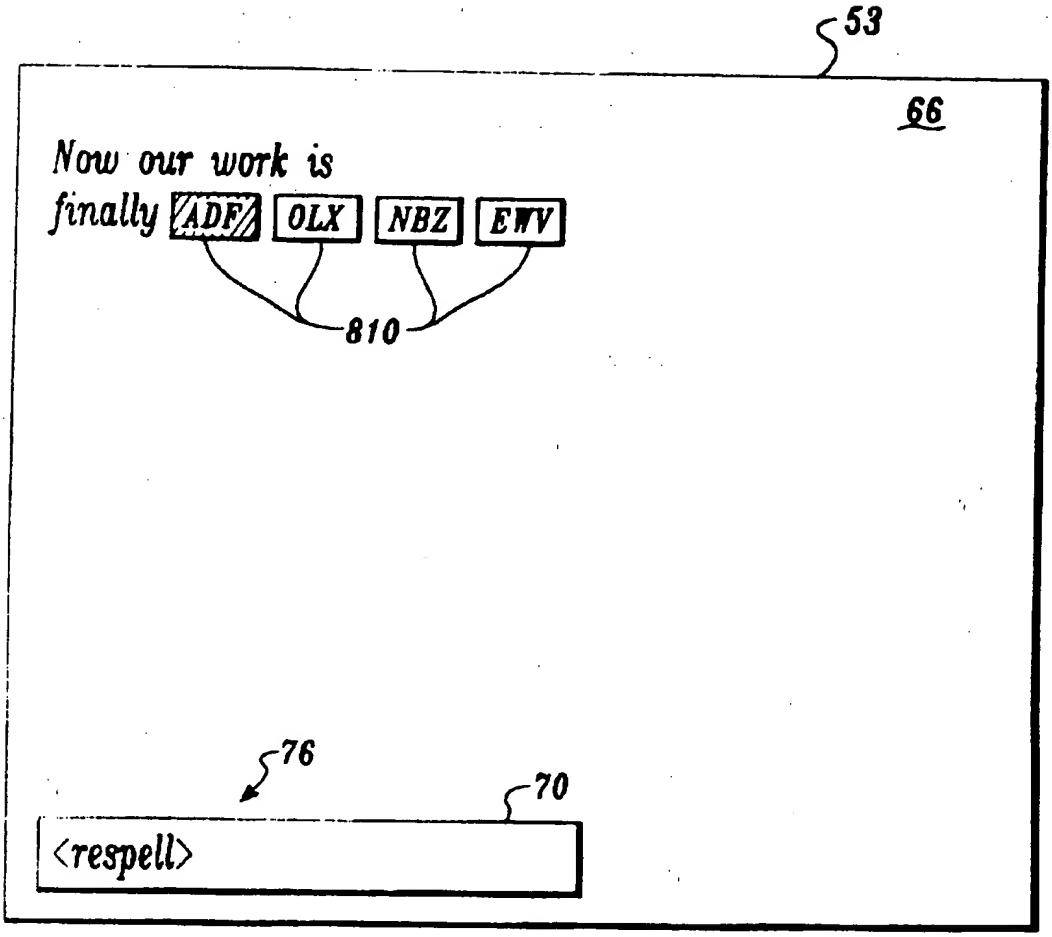


Fig. 7.

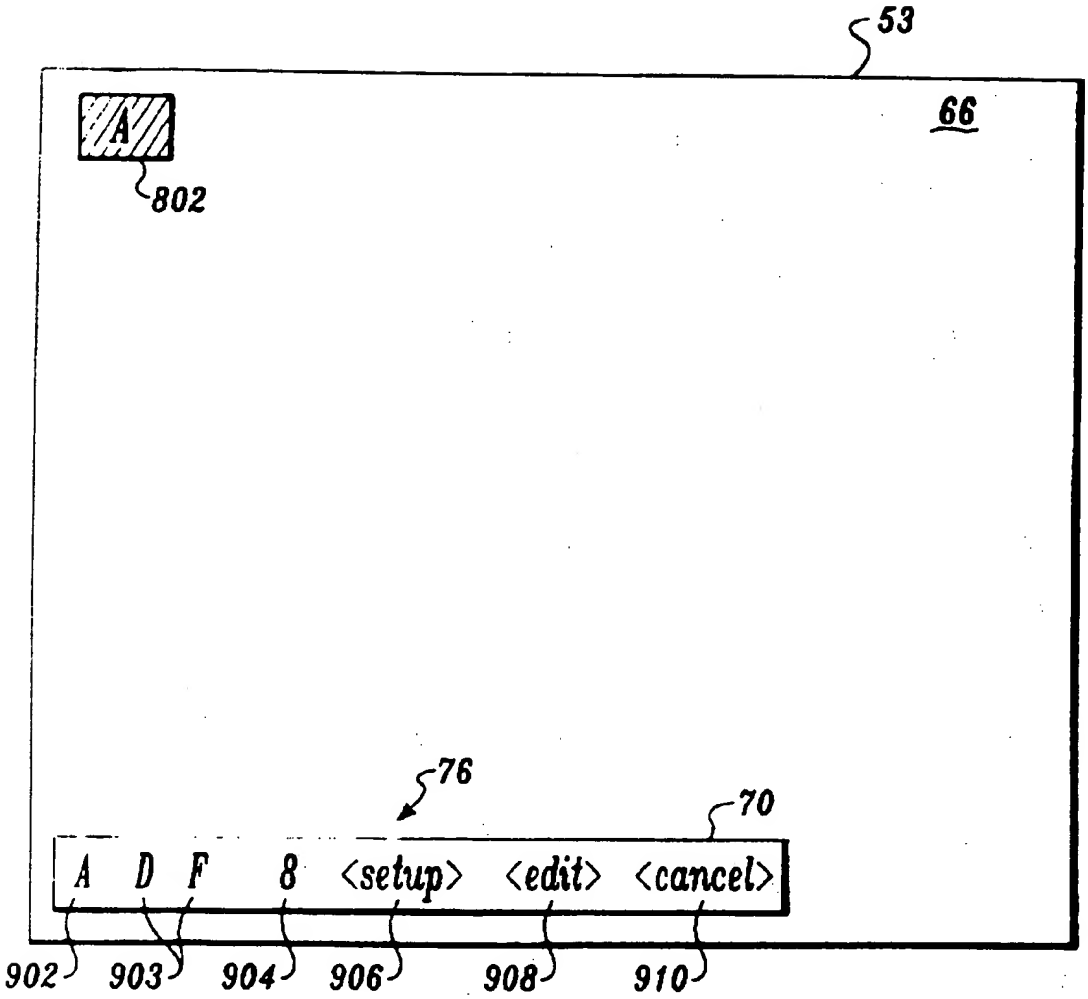


Fig. 8A.

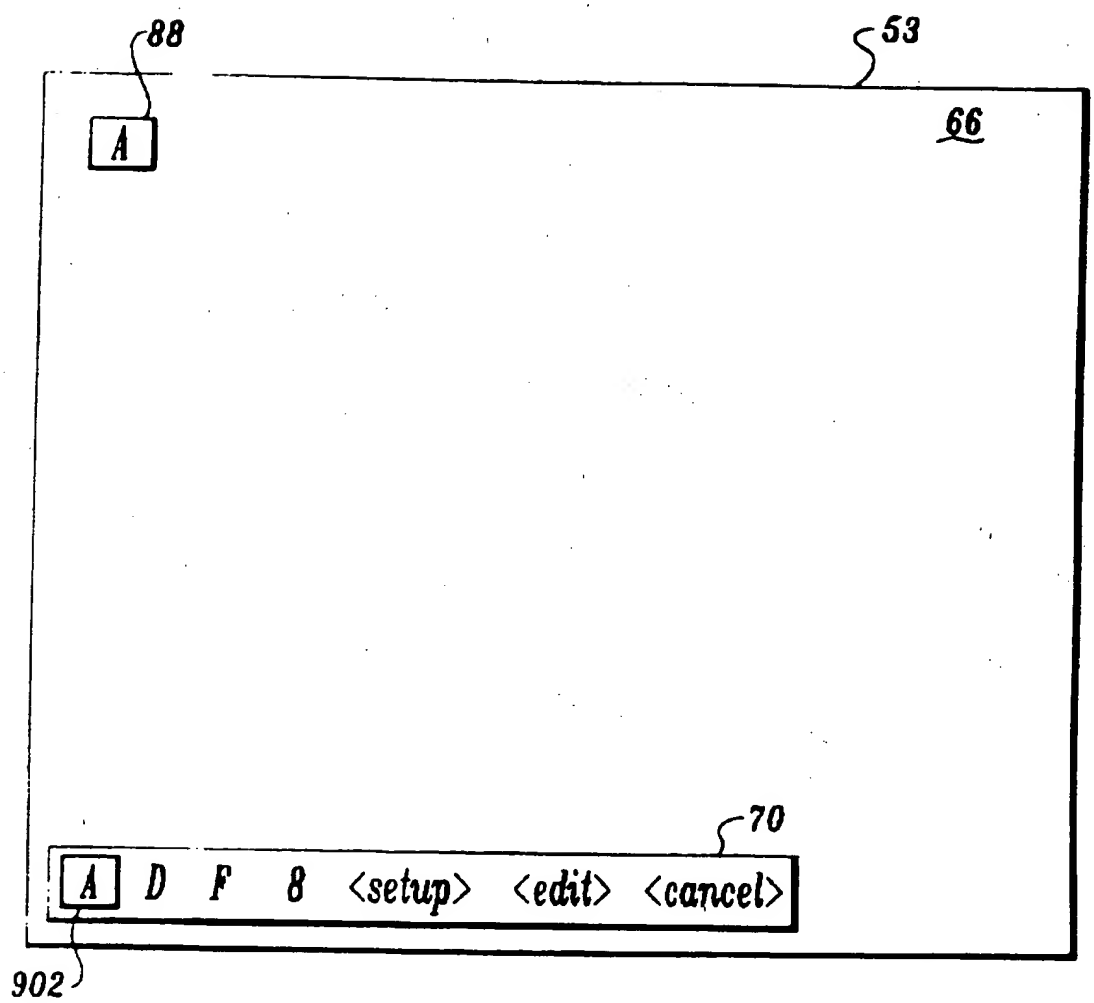


Fig. 8B.

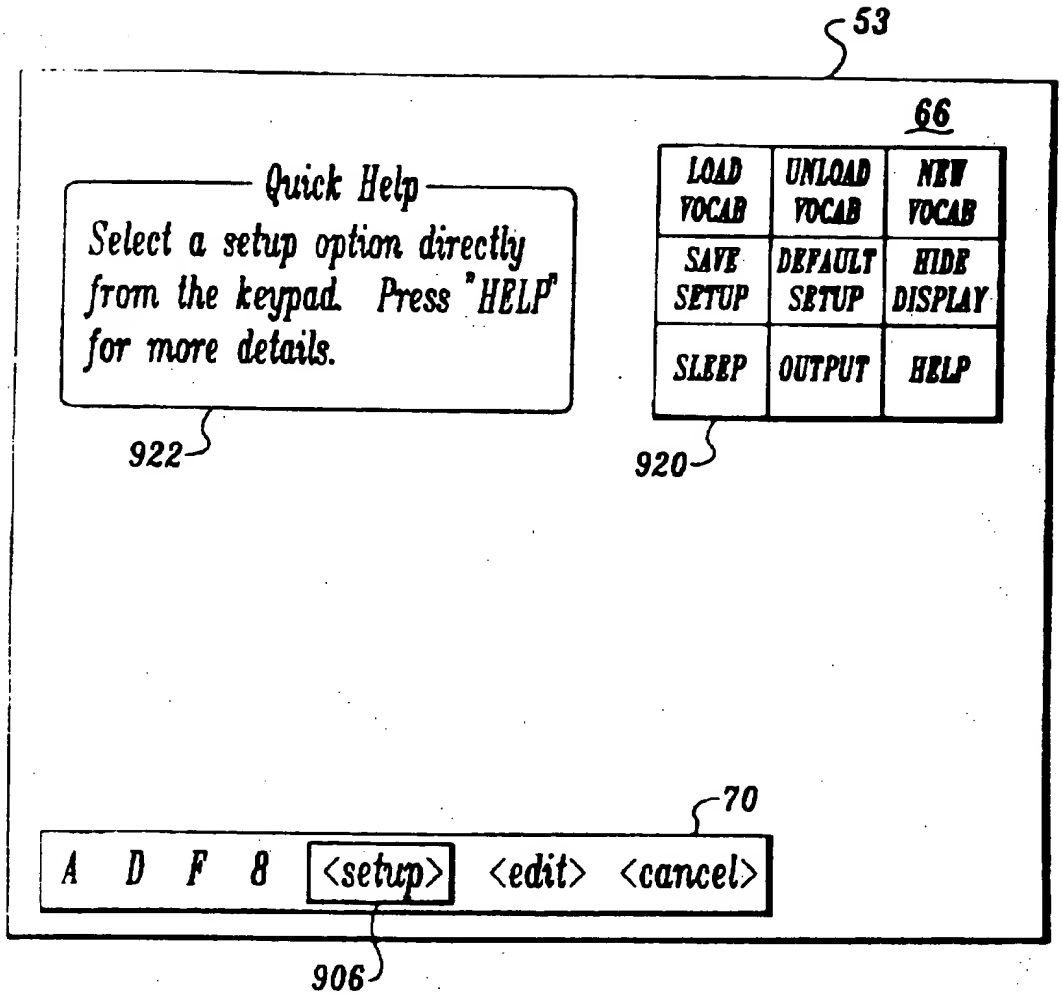

53
66
Quick Help
Select a setup option directly from the keypad. Press "HELP" for more details.
922
LOAD VOCAB
UNLOAD VOCAB
NEW VOCAB
SAVE SETUP
DEFAULT SETUP
HIDE DISPLAY
SLEEP
OUTPUT
HELP
920
70
A D F 8 <setup> <edit> <cancel>
906

Fig. 8C.

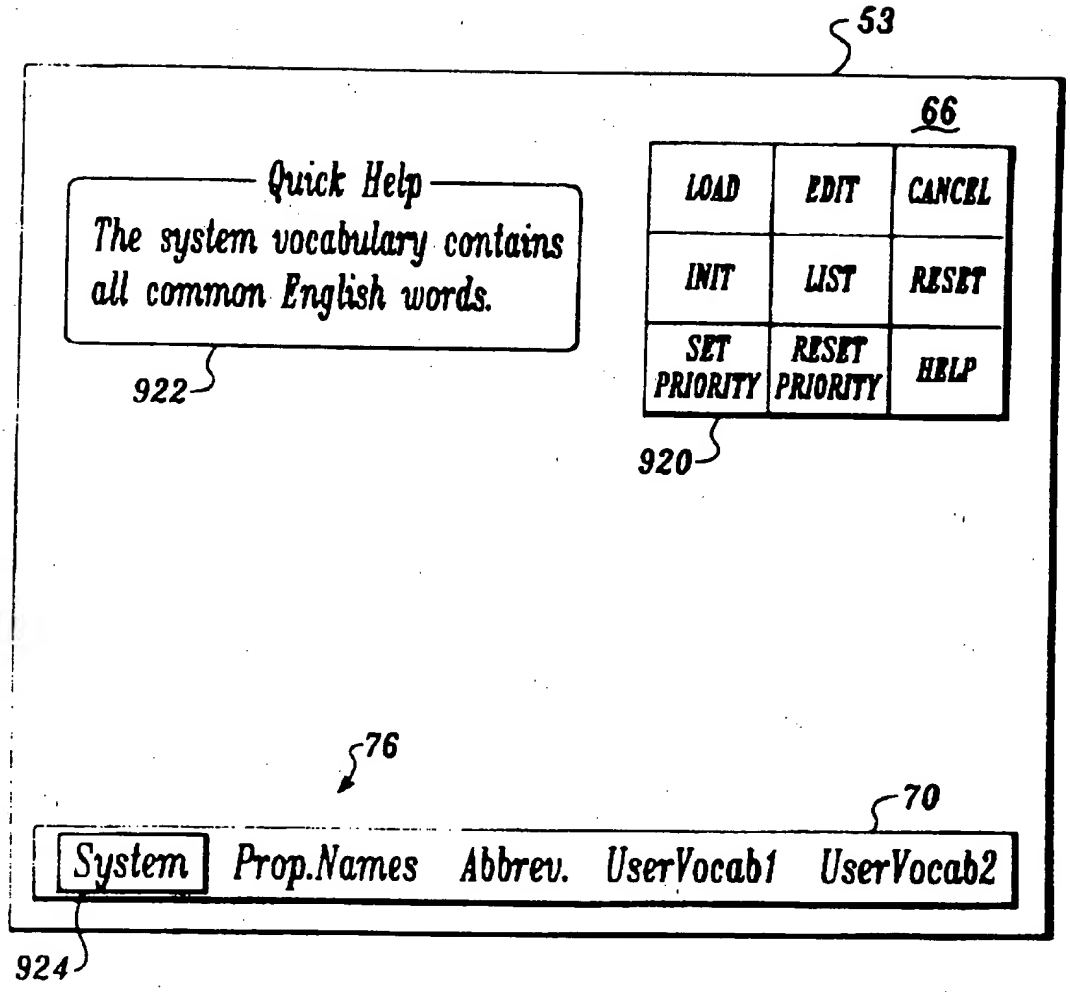

53
66
Quick Help
The system vocabulary contains all common English words.
922
LOAD
EDIT
CANCEL
INIT
LIST
RESET
SET PRIORITY
RESET PRIORITY
HELP
920
76
70
System
Prop.Names
Abbrev.
UserVocab1
UserVocab2
924

Fig. 8D.

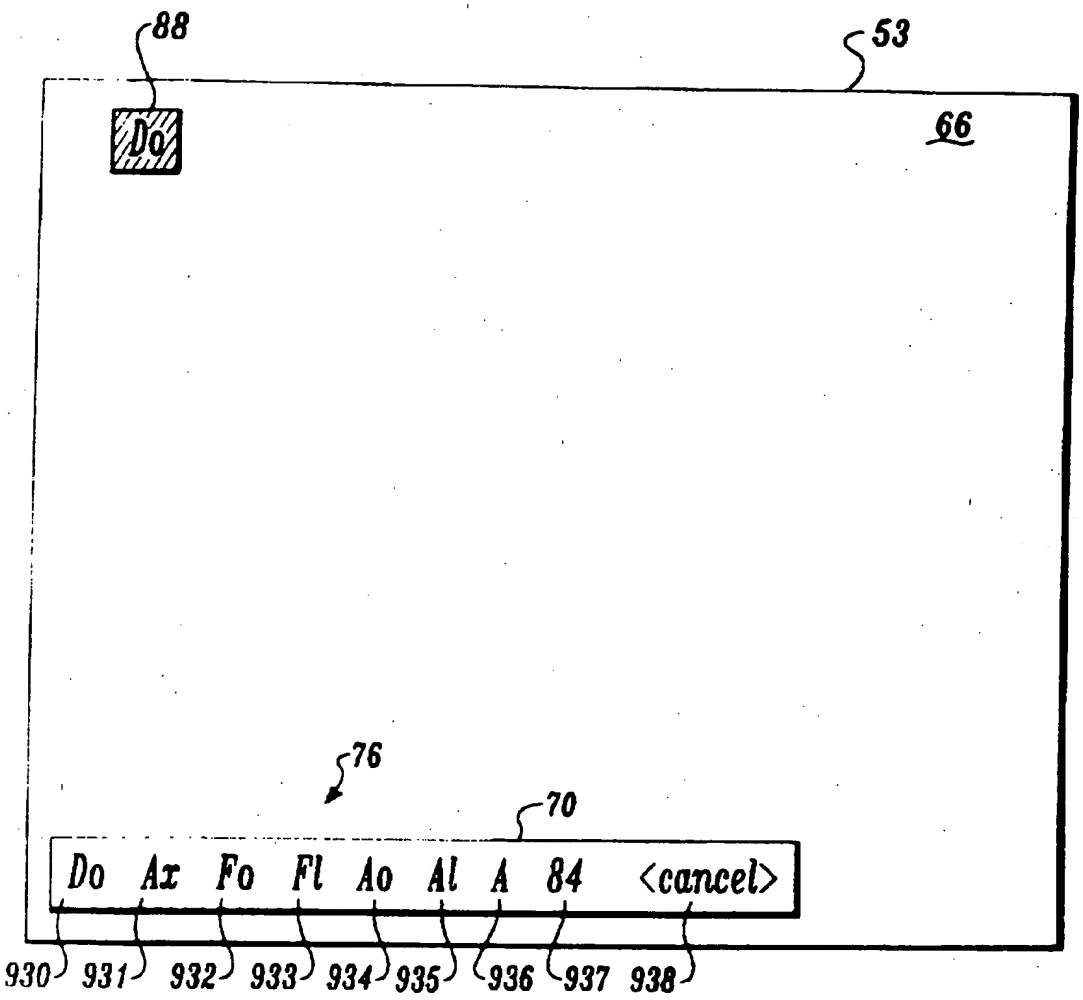

88
53
Do
66
76
70
Do Ax Fo Fl Ao Al A 84 <cancel>
930 931 932 933 934 935 936 937 938

Fig. 8E.

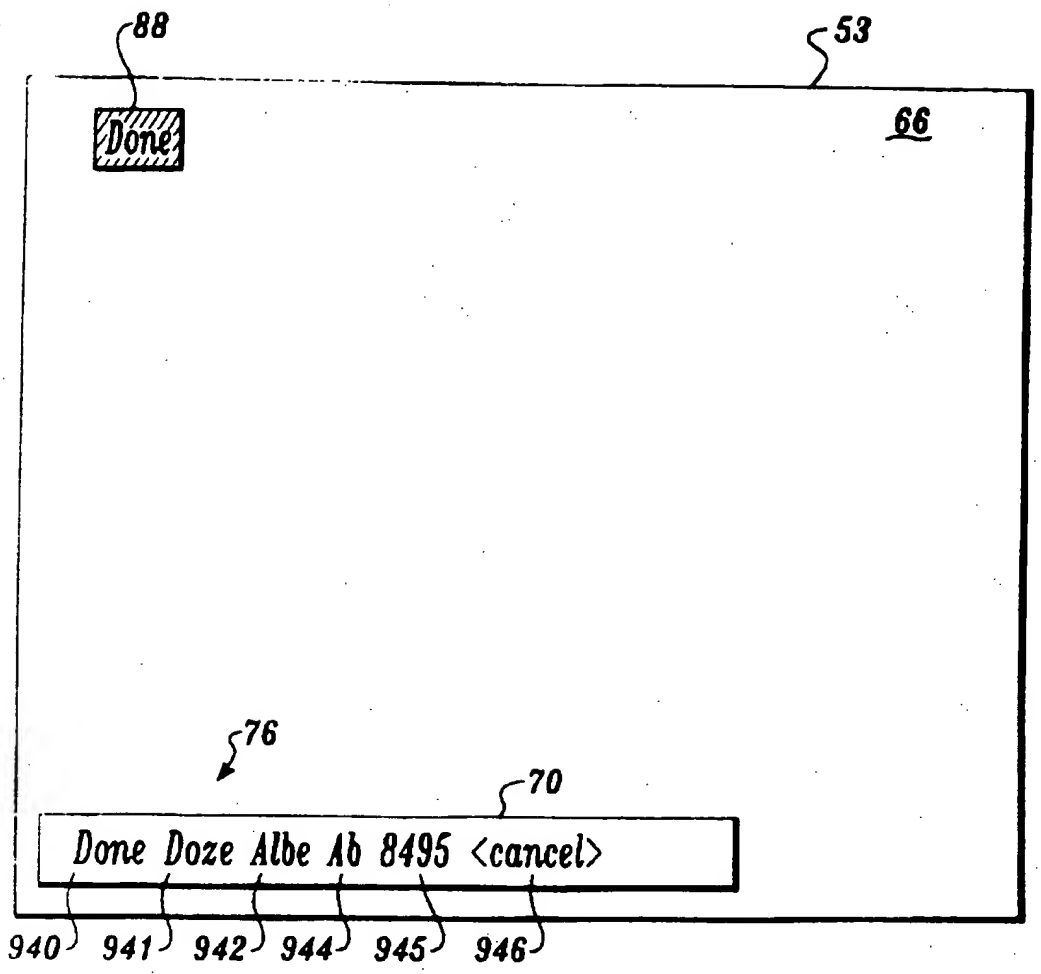

88
53
Done
66
76
70
Done Doze Albe Ab 8495 <cancel>
940
941
942
944
945
946

Fig. 8F.

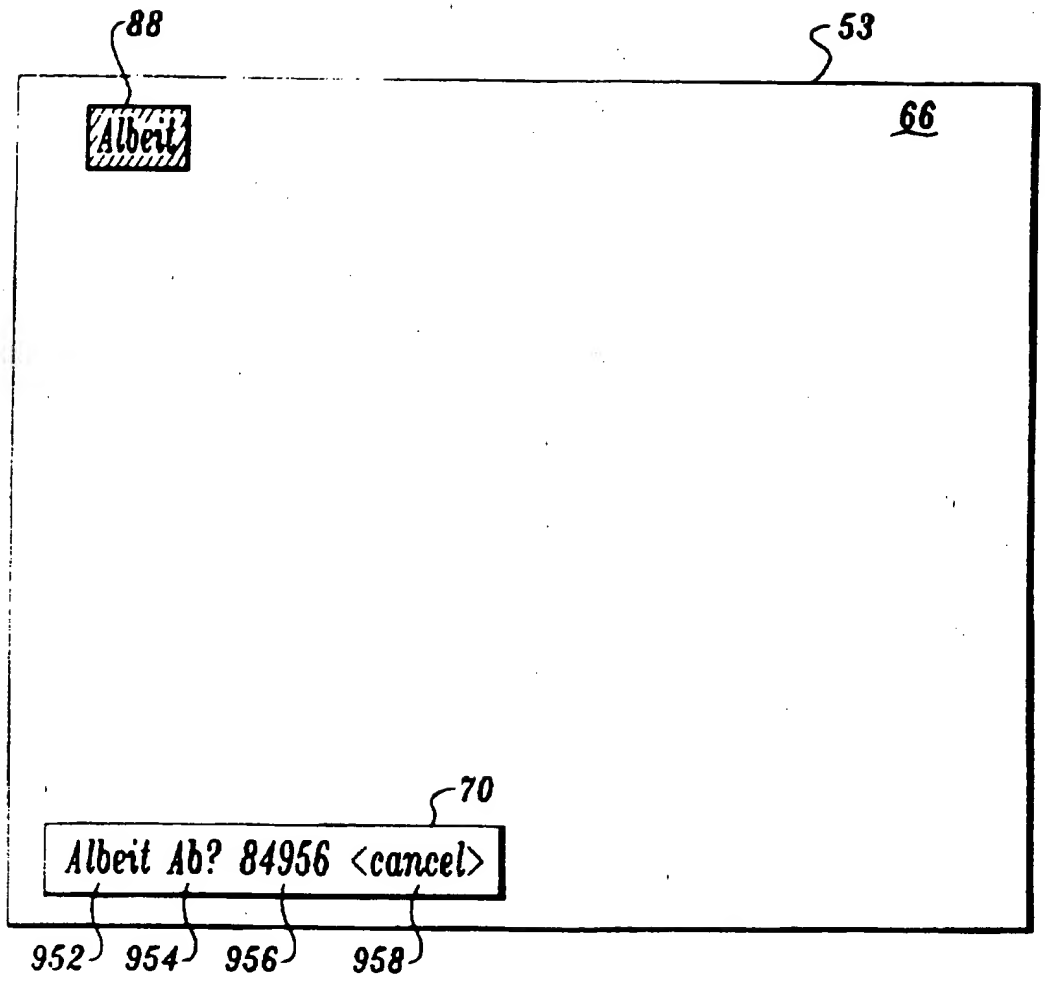


Fig. 8G.

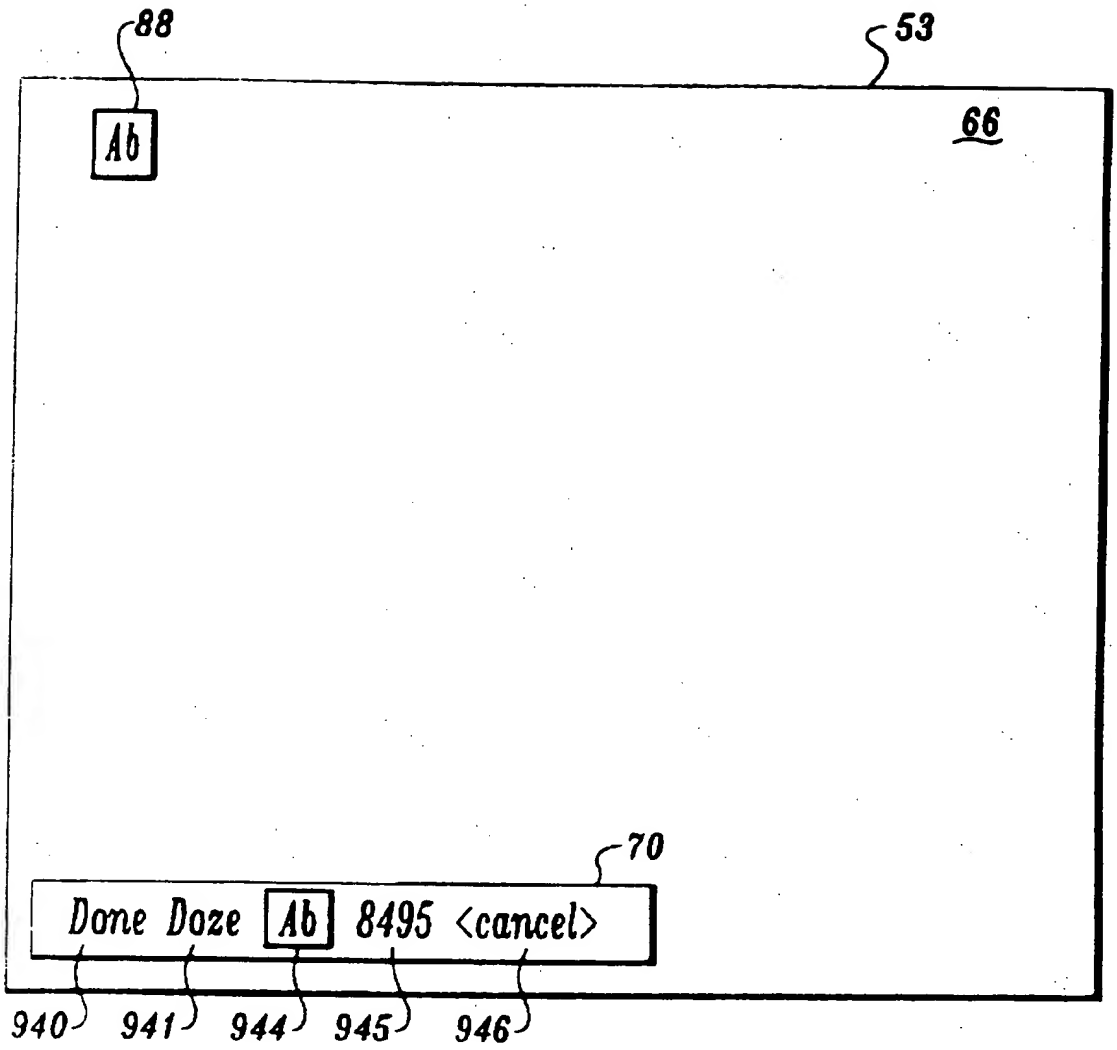


Fig. 8H.

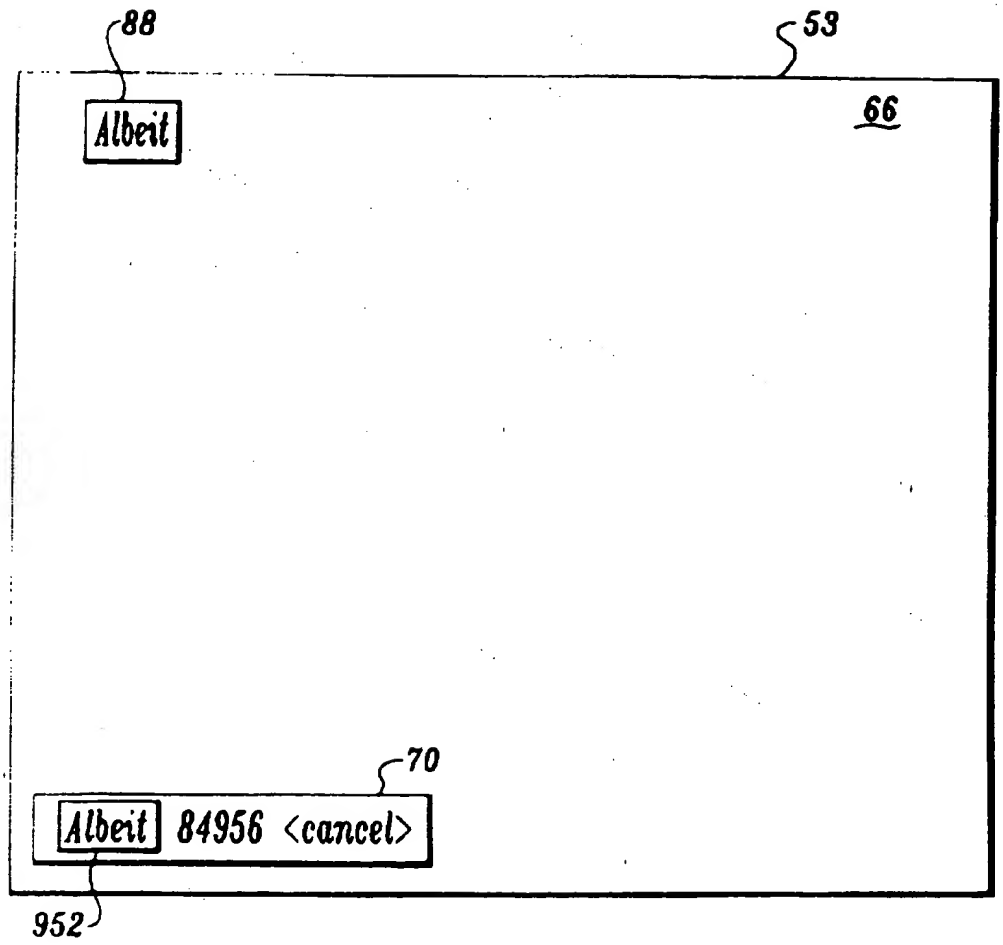


Fig. 8J

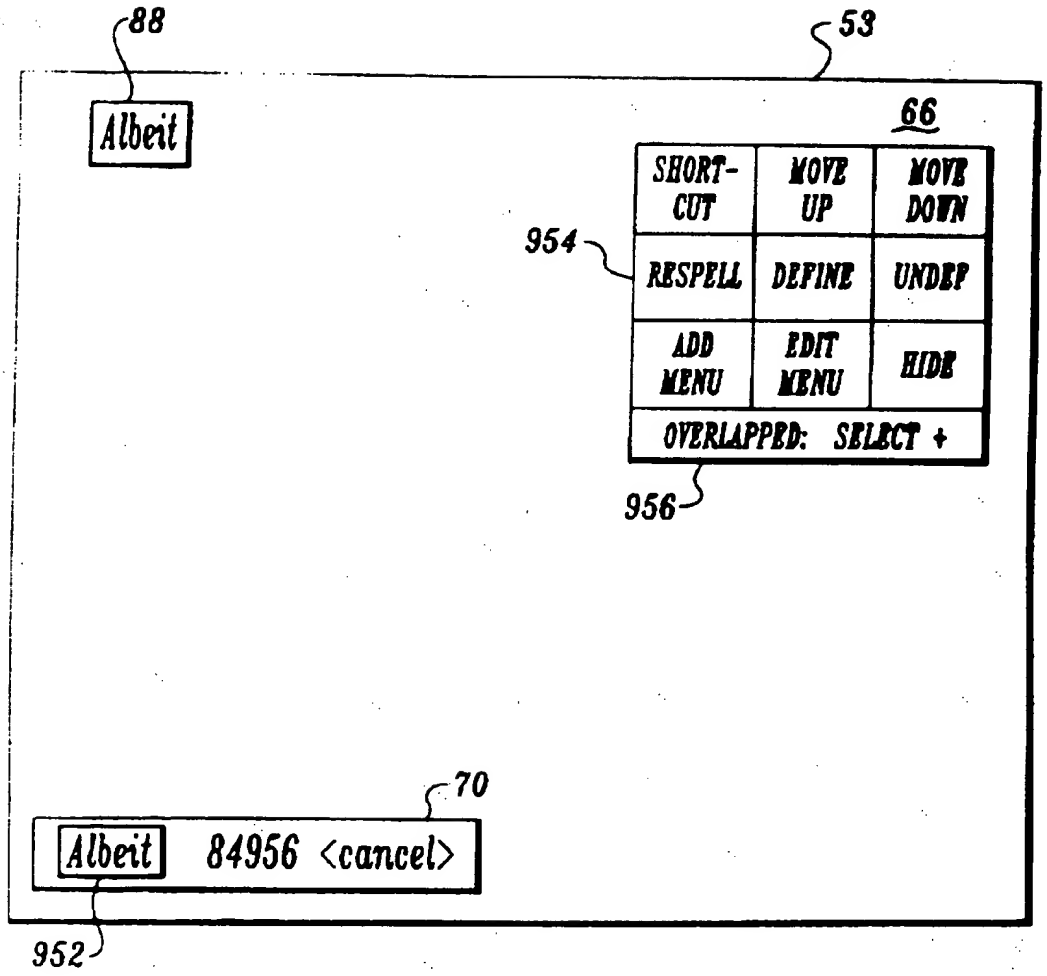

88
53
Albeit
66
SHORT-CUT
MOVE UP
MOVE DOWN
954
RESPELL
DEFINE
UNDEF
ADD MENU
EDIT MENU
HIDE
OVERLAPPED: SELECT +
956
70
Albeit
84956 <cancel>
952

Fig8J.

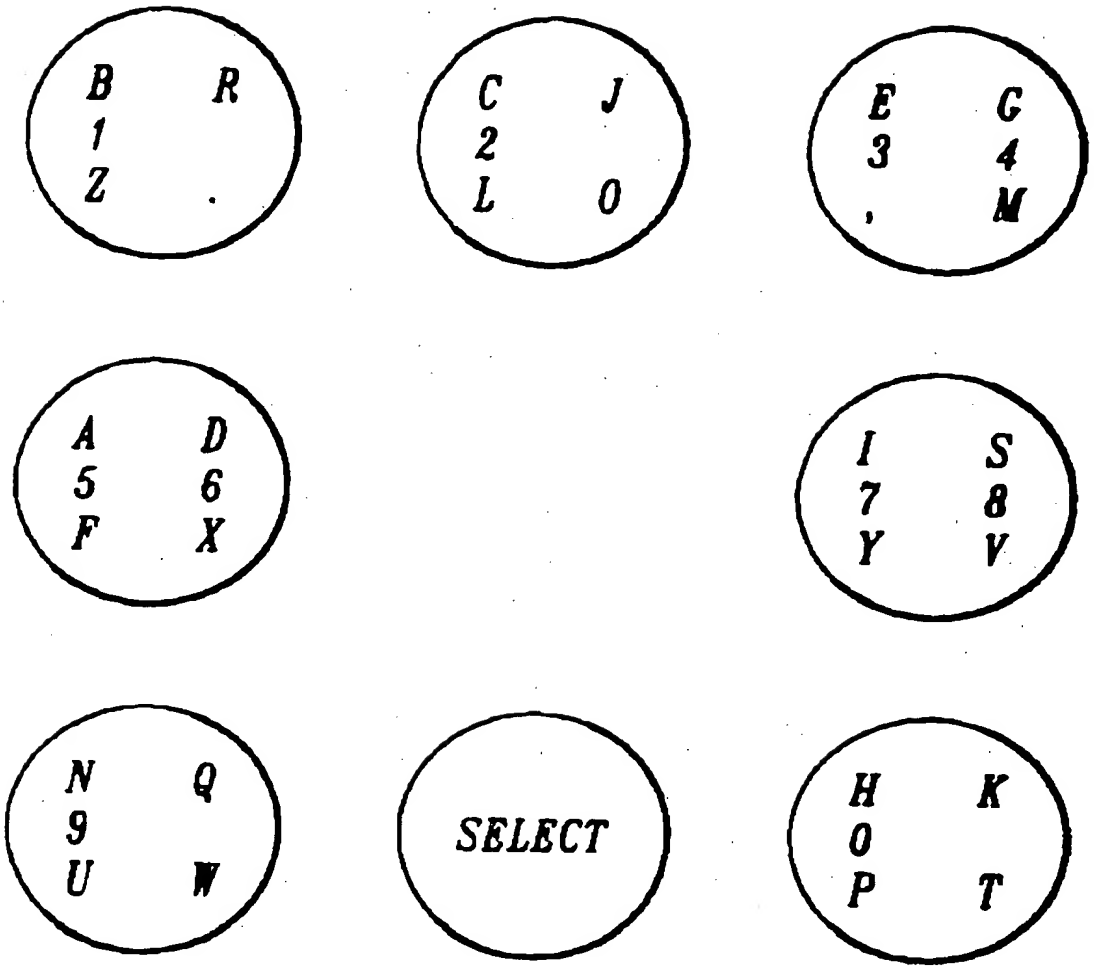


Fig. 9.

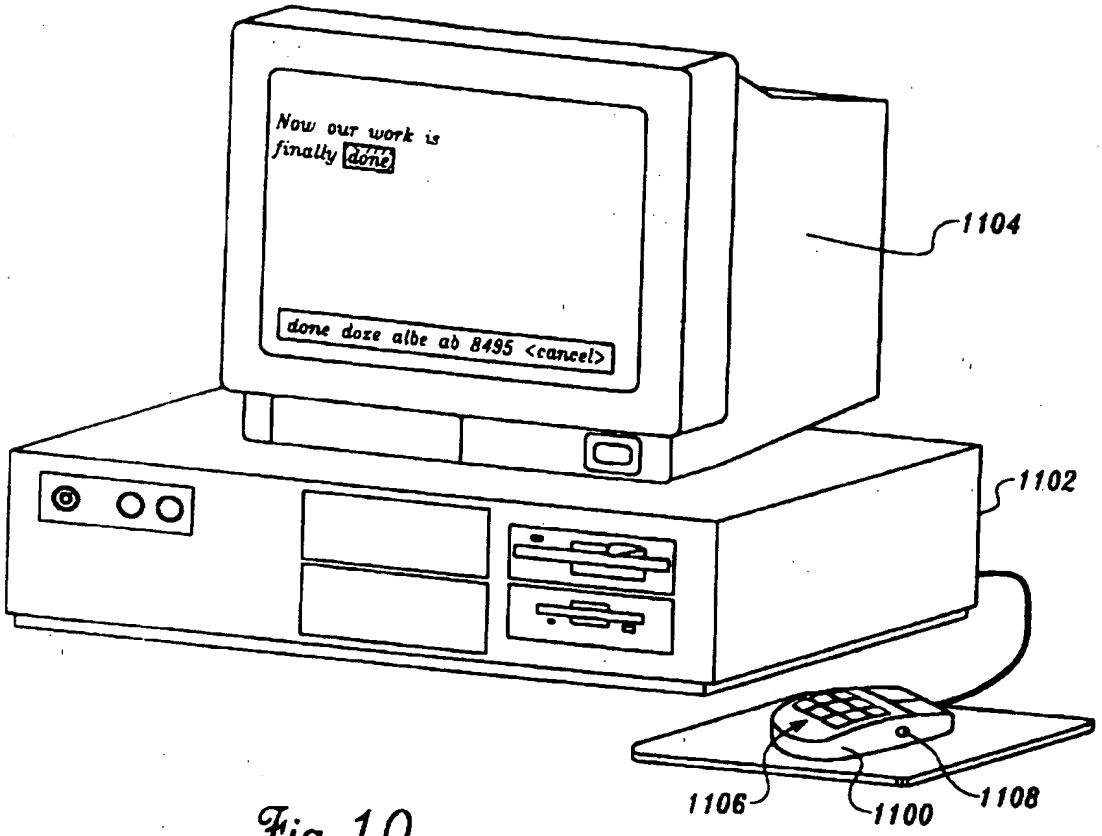

Now our work is
finally done
done doze albe ab 8495 <cancel>
1104
1102
1106
1100
1108

Fig. 10.

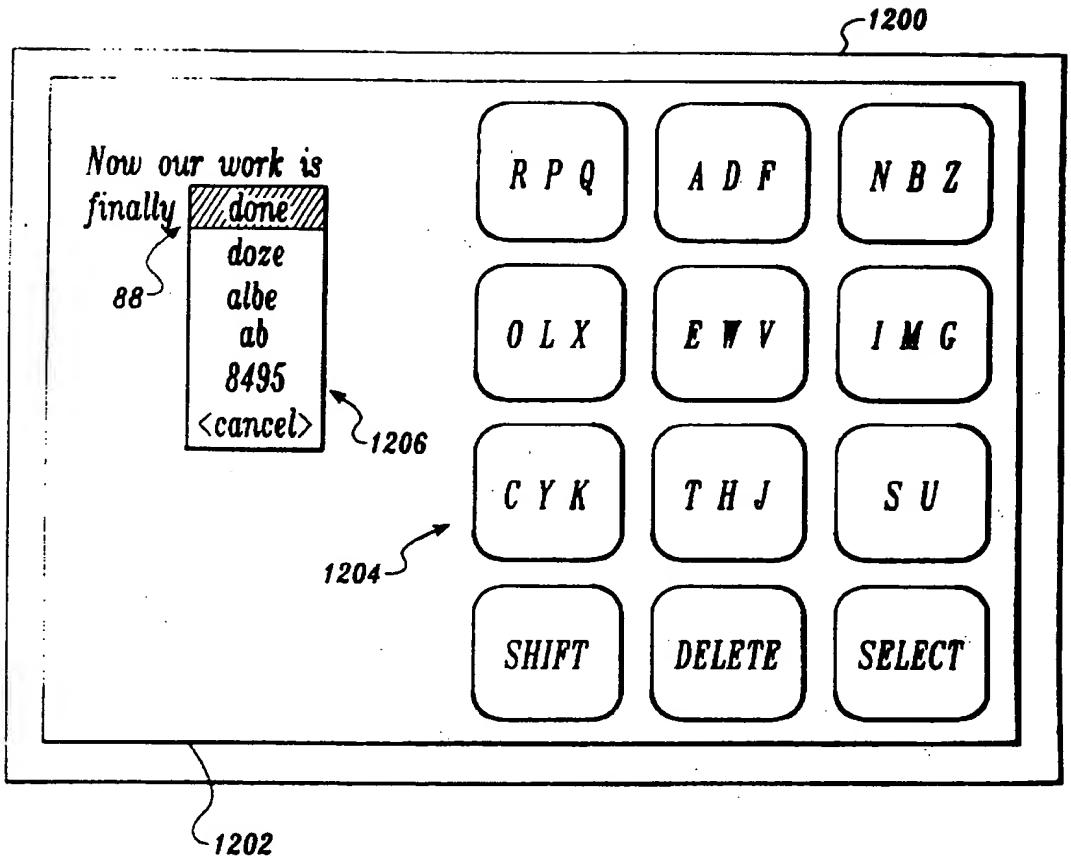

1200
Now our work is
finally
done
doze
albe
ab
8495
<cancel>
88
1206
1204
1202
R P Q
A D F
N B Z
O L X
E W V
I M G
C Y K
T H J
S U
SHIFT
DELETE
SELECT

Fig. 11.

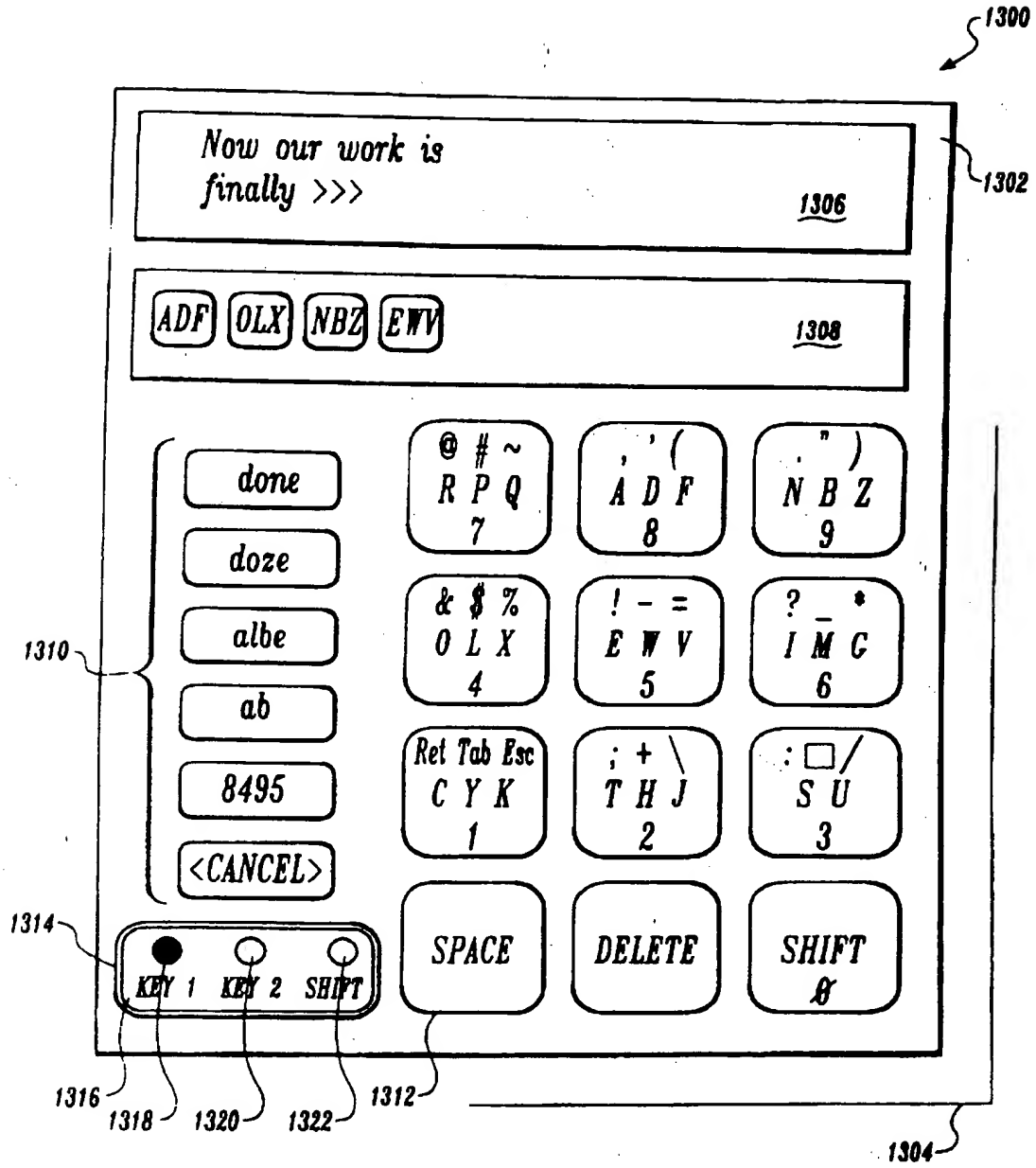

1300
Now our work is finally >>>
1306
1302
ADF OLX NBZ EWV
1308
done
doze
albe
ab
8495
<CANCEL>
1310
@ # ~ R P Q 7
, ' (A D F 8
") N B Z 9
& $ % O L X 4
! - = E W V 5
? _ * I M C 6
Ret Tab Esc C Y K 1
; + \ T H J 2
: □ / S U 3
SPACE
DELETE
SHIFT
1314
KEY 1
KEY 2
SHIFT
1316
1318
1320
1322
1312
1304

Fig. 12.

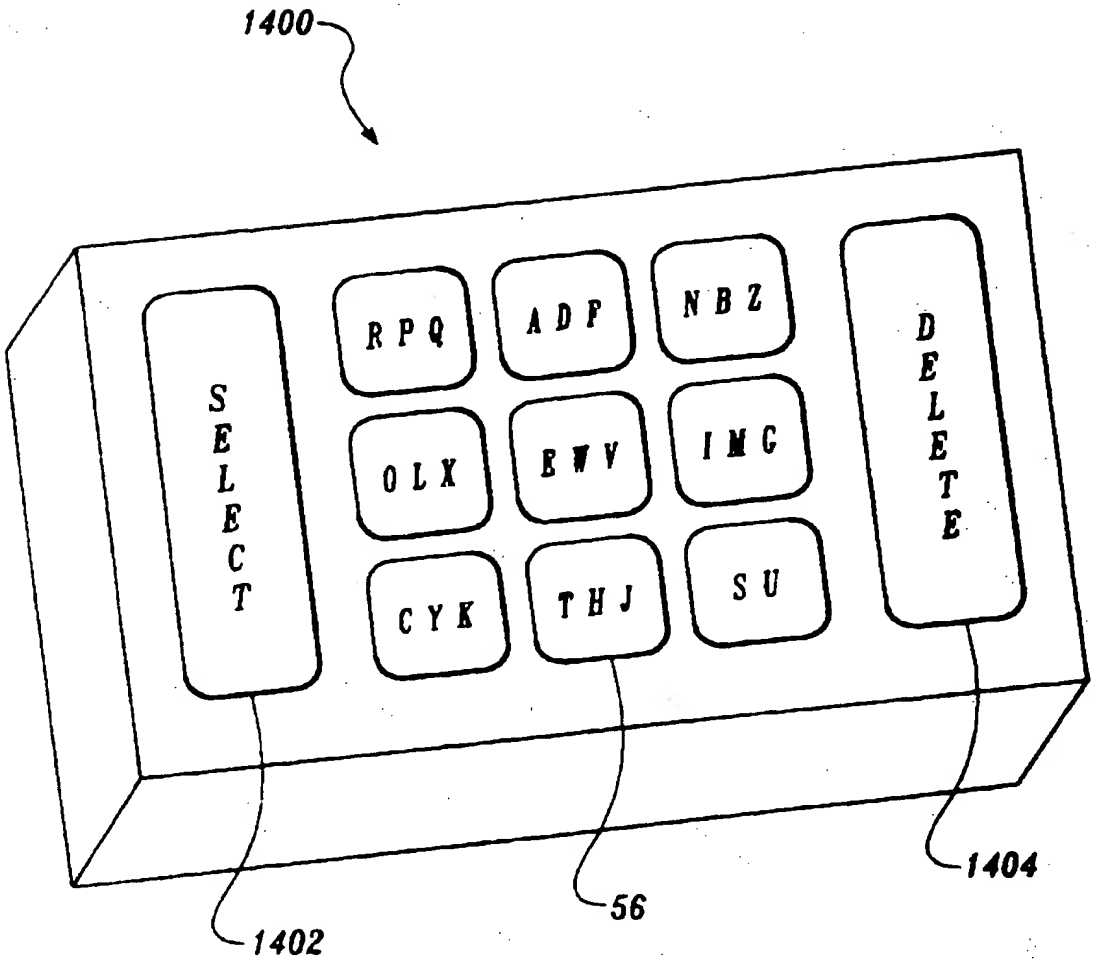

1400
SELECT
RPQ
ADF
NBZ
OLX
EWV
IMG
CYK
THJ
SU
DELETE
1402
56
1404

Fig. 13.

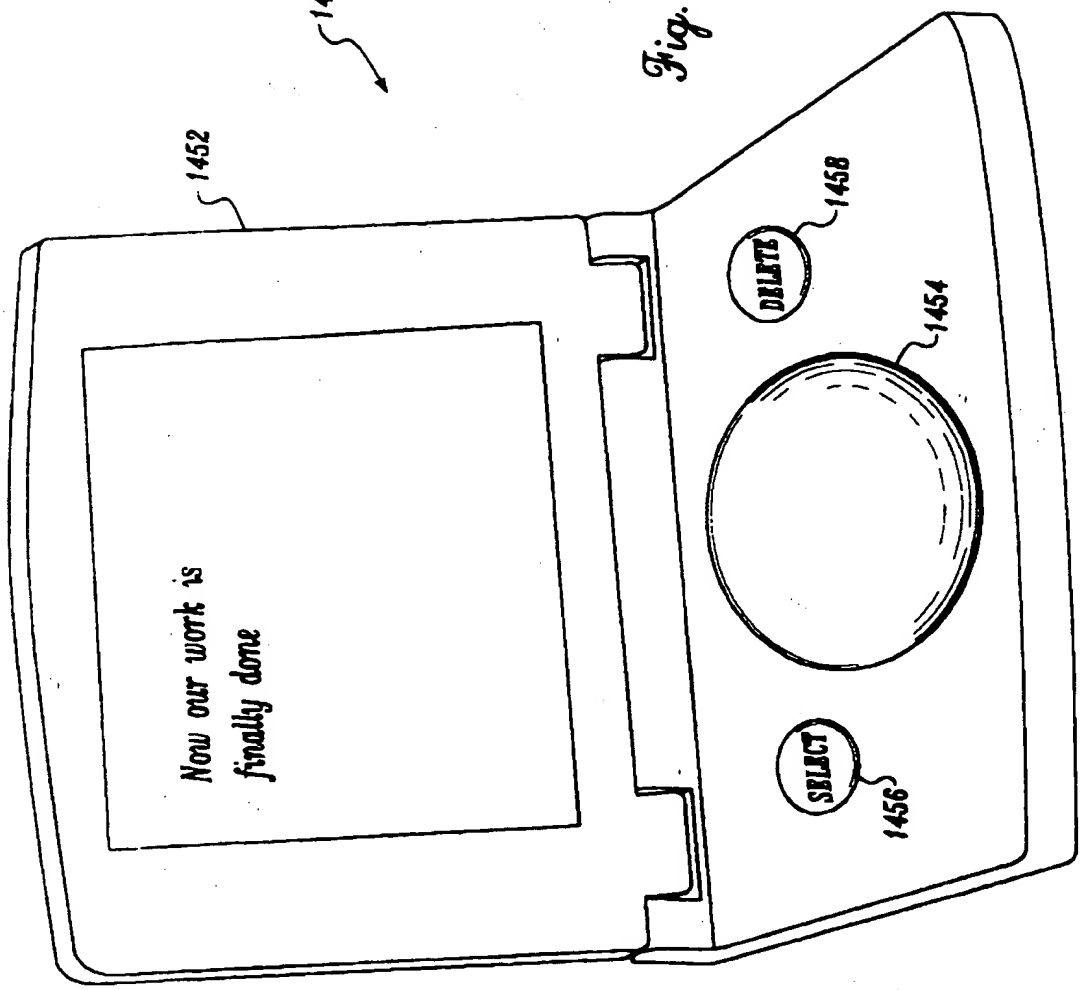

1450
1452
Now our work is finally done
DELETE
1458
1454
SELECT
1456

Fig. 14.

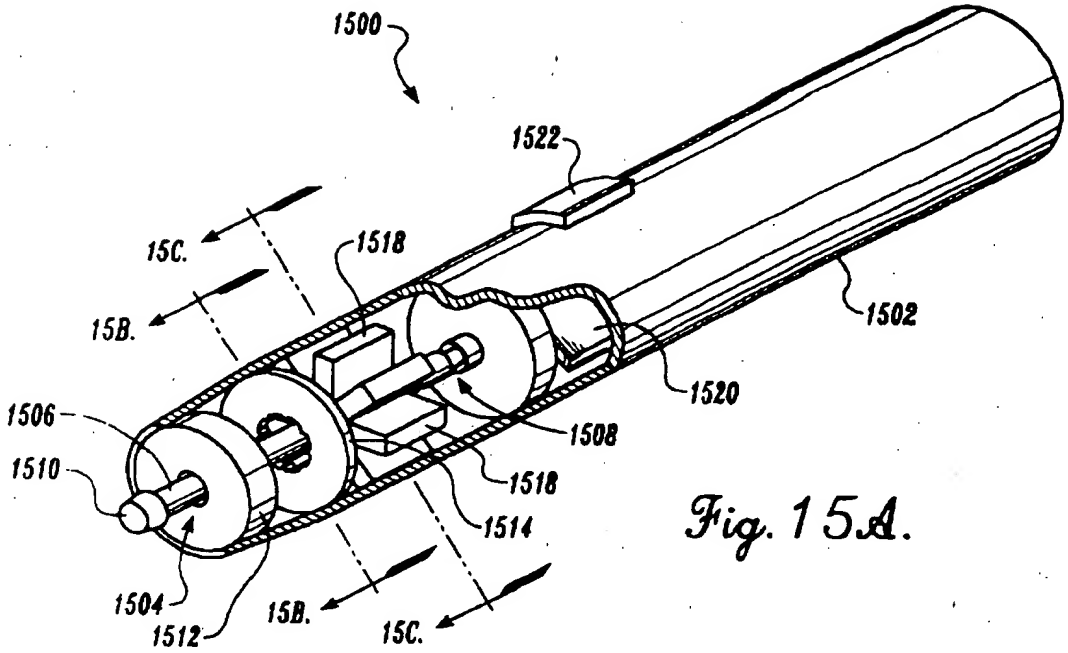


Fig. 15A.

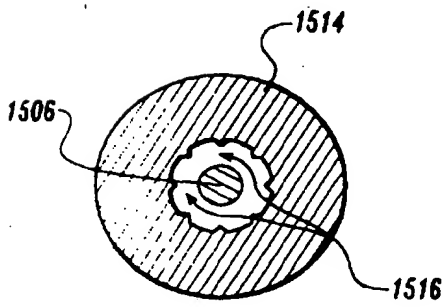


Fig. 15B.

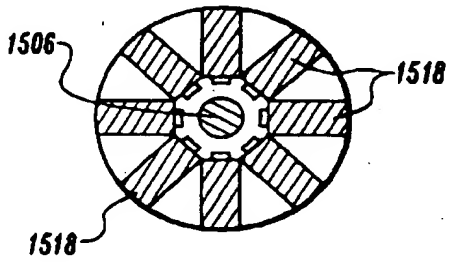


Fig. 15C.

REDUCED KEYBOARD DISAMBIGUATING SYSTEM

RELATIONSHIP OF OTHER APPLICATIONS

This application is a continuation-in-part of U.S. patent application Ser. No. 08/507,756, filed Jul. 26, 1995, which issued on Oct. 6, 1998 as U.S. Pat. No. 5,818,437 and U.S. provisional application Ser. No. 60/021,180, filed Jun. 10, 1996 pending, the benefit of the filing of which is hereby claimed under 35 U.S.C. §120.

FIELD OF THE INVENTION

The invention relates generally to reduced keyboard systems, and more specifically to reduced keyboard systems using disambiguation to resolve ambiguous keystrokes.

BACKGROUND OF THE INVENTION

For many years, portable computers have been getting smaller and smaller. The principal size-limiting component in the effort to produce a smaller portable computer has been the keyboard. If standard typewriter-size keys are used, the portable computer must be at least as large as the keyboard. Miniature keyboards have been used on portable computers, but the miniature keyboard keys have been found to be too small to be easily or quickly manipulated by a user.

Incorporating a full-size keyboard in a portable computer also hinders true portable use of the computer. Most portable computers cannot be operated without placing the computer on a flat work surface to allow the user to type with both hands. A user cannot easily use a portable computer while standing or moving. In the latest generation of small portable computers, called Personal Digital Assistants (PDAs), companies have attempted to address this problem by incorporating handwriting recognition software in the PDA. A user may directly enter text by writing on a touch-sensitive panel or screen. Unfortunately, the accuracy and speed of the handwriting recognition software has to date been less than satisfactory.

It would therefore be advantageous to develop a keyboard for entry of text into a computer that is both small and operable with one hand while the user is holding the computer with the other hand. Prior development work has considered use of a keyboard that has a reduced number of keys. As suggested by the keypad layout of a Touch-Tone telephone, many of the reduced keyboards have used a 3-by-4 array of keys. Each key in the array of keys contains multiple characters. As a user enters a sequence of keys, there is therefore ambiguity in the resulting output since each keystroke may indicate one of several letters. Several approaches have been suggested for resolving the ambiguity of the keystroke sequence.

One suggested approach for unambiguously specifying characters entered on a reduced keyboard requires the user to enter two or more keystrokes to specify each letter. The keystrokes may be entered either simultaneously (chording) or in sequence (two-stroke specification). Neither chording nor two-stroke specification has produced a keyboard having adequate simplicity and efficiency of use. Two-stroke specification is inefficient, and chording is complicated to learn and use.

Other suggested approaches for determining the correct character sequence that corresponds to an ambiguous keystroke sequence are summarized in the article "Probabilistic Character Disambiguation for Reduced Keyboards Using Small Text Samples," published in the Journal of the International Society for Augmentative and Alternative Communication by John L. Arnott and Muhammad Y. Javad (hereinafter the "Arnott article"). The Arnott article notes that the majority of disambiguation approaches employ known statistics of character sequences in the relevant language to resolve character ambiguity in a given context. That is, existing disambiguating systems statistically analyze ambiguous keystroke groupings as they are being entered by a user to determine the appropriate interpretation of the keystrokes. The Arnott article also notes that several disambiguating systems have attempted to use word level disambiguation to decode text from a reduced keyboard. Word level disambiguation disambiguates entire words by comparing the sequence of received keystrokes with possible matches in a dictionary after the receipt of an unambiguous character signifying the end of the word. The Arnott article discusses many of the disadvantages of word-level disambiguation. For example, word level disambiguation oftentimes fails to decode a word correctly, because of the limitations in identifying unusual words and the inability to decode words that are not contained in the dictionary. Because of the decoding limitations, word level disambiguation does not give error-free decoding of unconstrained English text with an efficiency of one keystroke per character. The Arnott article therefore concentrates on character level disambiguation rather than word level disambiguation, and indicates that character level disambiguation appears to be the most promising disambiguation technique.

Still another suggested approach is disclosed in a textbook entitled *Principles of Computer Speech*, authored by I. H. Witten, and published by Academic Press in 1982 (hereinafter the "Witten approach"). Witten discusses a system for reducing ambiguity from text entered using a telephone touch pad. Witten recognizes that for approximately 92% of the words in a 24,500 word dictionary, no ambiguity will arise when comparing the keystroke sequence with the dictionary. When ambiguities do arise, however, Witten notes that they must be resolved interactively by the system presenting the ambiguity to the user and asking the user to make a selection between the number of ambiguous entries. A user must therefore respond to the system's prediction at the end of each word. Such a response slows the efficiency of the system and increases the number of keystrokes required to enter a given segment of text.

Disambiguating an ambiguous keystroke sequence continues to be a challenging problem. As noted in the publications discussed above, satisfactory solutions that minimize the number of keystrokes required to enter a segment of text have failed to achieve the necessary efficiencies to be acceptable for use in a portable computer. Moreover, the publications have taught away from using word level disambiguation and have focused on character level disambiguating techniques. It would therefore be desirable to develop a disambiguating system that minimizes the ambiguity of entered keystrokes and thereby maximizes the efficiency of text entry.

SUMMARY OF THE INVENTION

The present invention provides a reduced keyboard using word level disambiguation to resolve ambiguities in keystrokes. The keyboard is constructed with full-size keys, preferably twelve keys arrayed in three columns and four rows or three rows and four columns. A plurality of letters and symbols are assigned to at least several of the keys, so that keystrokes by a user are ambiguous. A user enters a keystroke sequence wherein each keystroke corresponds to the entry of one letter of a word. Because individual key-

strokes are ambiguous, the keystroke sequence could potentially match more than one word with the same number of letters. The keystroke sequence is processed by comparing the keystroke sequence with stored vocabulary modules to match the sequence with corresponding stored words or other interpretations. Words that match the sequence of keystrokes are simultaneously and automatically presented to the user in a list on a display as each keystroke is received (hereinafter referred to as "the selection list"). The other interpretations of the keystroke sequence are also provided to the user in the selection list. The words and other interpretations are presented in order of decreasing frequency of use, with the most commonly used word presented first.

In accordance with one aspect of the invention, the user presses an unambiguous select key to delimit an entered keystroke sequence. After receiving the select key, the disambiguating system automatically selects the most commonly used word and adds the word to the sentence being constructed if the user continues to enter additional text.

In accordance with another aspect of the invention, the select key that is pressed by the user to delimit the end of a word is also used to select less commonly used words from the selection list presented to the user. If the most commonly used word presented to the user at the top of the selection list is not the desired word, the user presses the select key again to advance from the most frequently used word to the second most frequently used word, and again to advance to the third most frequently used word, and so on. By repetitively pressing the select key, the user may therefore select the desired word from the selection list. Upon reaching the end of the displayed selection list, additional less commonly used words scroll into the selection list with continued presses of the select key. The reduced keyboard disambiguating system automatically inserts the appropriate spacing between the words.

In accordance with yet another aspect of the invention, multiple interpretations of the keystroke sequence are provided to the user in the selection list. The keystroke sequence is interpreted as forming a word, and the corresponding word is displayed in the selection list. Simultaneously, the keystroke sequence is interpreted as a number, as a word entered using a two-stroke or multiple-stroke specification method, as a stem of an uncompleted word, or as a system command. The multiple interpretations are simultaneously presented to the user for each keystroke of a keystroke sequence entered by the user. The user may select from the alternate interpretations by pressing the select key a number of times.

In accordance with still another aspect of the invention, a method to enter words that are not contained in the vocabulary is provided. To enter unusual words, such as proper names, a two-stroke specification method is used. A first keystroke indicates the key containing the letter, and a second keystroke disambiguates the first keystroke. Preferably, the system simultaneously interprets each sequence of keystrokes as both an ambiguous sequence of one stroke per letter and as a fully specified sequence of two strokes per letter.

In accordance with still another aspect of the invention, a second method to enter unusual words that are not contained in the vocabulary is provided, using a multiple-stroke specification method. In this method, a sequence of keystrokes is interpreted as unambiguously specifying a specific string of alphabetic characters by requiring from one to three keypresses per letter. Preferably, each key contains up to three letters that are arranged in a row on the top of each key. The letters are ordered from left to right on each keytop in order of decreasing frequency, so that the most frequently occurring letter is in the leftmost position. Each letter of a word may be unambiguously spelled by pressing the key on which the letter appears a number of times corresponding to the position of the desired letter in the row. Thus, the leftmost letter is specified by a single keypress, the center letter by two keypresses, and the rightmost letter by three keypresses. The system simultaneously interprets each sequence of keystrokes as both an ambiguous sequence of one stroke per letter and as a fully specified sequence using the multiple-stroke specification method.

At the user's option, either or both of the two-stroke and multiple-stroke interpretations are displayed in the selection list of possible words. The user therefore does not have to change modes to enter words using either of these specification methods. Instead, when the user wants to enter a word that is not in the vocabulary, the user simply uses the preferred specification method, and then presses the select key to move down the selection list and select the desired interpretation of the keystroke sequence. In the vast majority of cases, these interpretations will be the only valid choices left in the selection list, so that only one or two presses of the select key will be required.

In accordance with yet another aspect of the invention, words that are entered by the user using the two-stroke or multiple-stroke specification methods that are not in a vocabulary module are automatically added to a vocabulary module. The added words can subsequently be displayed upon entry of one stroke per letter.

In accordance with still another aspect of the invention, words are stored in a vocabulary module using a tree data structure. Words corresponding to a particular keystroke sequence are constructed using the set of words associated with an immediately preceding keystroke sequence (i.e., the particular keystroke sequence without the last keystroke). Constructing words in this manner greatly reduces the storage space of the vocabulary module.

In accordance with yet another aspect of the invention, letters are assigned to the keys in a non-sequential order. Rather than assigning the letters from "a" to "z," the letters are grouped onto the keys in a manner that reduces the frequency of ambiguities during word entry. In particular, the letters are assigned on the keys to take into account the frequency of use of each word in the disambiguating system's vocabulary modules. That is, the letters are grouped to reduce the ambiguities between the most commonly entered words. The grouping of the letters on the keys is therefore optimized with respect to one or more preferred vocabulary modules that include information regarding the frequency of common usage.

In accordance with still another aspect of the invention, the keyboard of the reduced keyboard disambiguating system may be replaced with an input device movable to multiple states. Each state is associated with a keystroke, allowing a user to input text by appropriate movement of the input device.

The combined effects of the non-sequential and optimized assignment of letters to keys, the delimiting of words using a select key, the presentation of the most commonly occurring word as the first word in the selection list, the inclusion of multiple interpretations in the selection list, the automatic addition of a selected word to a sentence by the first keystroke of the following word, and the automatic addition of spaces produces a surprising result: for over 99% of

entered words, the same number of keystrokes is required to enter a word with the reduced key disambiguating system as for word entry with a conventional keyboard. Because ambiguities are reduced and the words are presented in frequency of use order, the desired word is most often the first word presented and is frequently the only word presented. The user simply proceeds to enter the next word with no more than the usual number of keystrokes. High speed entry of text is therefore achieved using a keyboard having a small number of full-size keys.

Moreover, the reduced keyboard disambiguation system disclosed herein also minimizes the size of the computer or other device that incorporates the system. The reduced number of keys allows a device to be constructed that may be held by the user in one hand, while being operated with the other hand. The disclosed system is therefore particularly advantageous for use with PDAs, two-way pagers, or other small electronic devices that would benefit from accurate and high-speed text entry.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing aspects and many of the attendant advantages of this invention will become more readily appreciated as the same becomes better understood by reference to the following detailed description, when taken in conjunction with the accompanying drawings, wherein:

FIGS. 1A and 1B are perspective views of a preferred embodiment of a portable computer incorporating a reduced keyboard disambiguating system of the present invention, the disambiguating system including a reduced keyboard and a display for displaying text and a selection list;

FIG. 2 is a hardware block diagram of the reduced keyboard disambiguating system of FIG. 1;

FIG. 3 is a flow chart of disambiguating software for the reduced keyboard disambiguating system of FIG. 1;

FIGS. 4A–4C are diagrams depicting the construction of a preferred vocabulary module and associated object lists for the reduced keyboard disambiguating system of FIG. 1;

FIG. 5 is a flow chart of a subroutine for identifying objects contained in the vocabulary module depicted in FIG. 4 that correspond to a received keystroke sequence;

FIG. 6 depicts the display of the preferred embodiment of the reduced keyboard disambiguating system of the present invention wherein a key map for editing is displayed on the display;

FIG. 7 depicts the display of the preferred embodiment of the reduced keyboard disambiguating system of the present invention wherein a respell mode has been entered by a user;

FIGS. 8A–8J depict the display of the preferred embodiment of the reduced keyboard disambiguating system of the present invention during a representative use of the preferred disambiguation method;

FIG. 9 is a diagram of an eight-key reduced keyboard layout for individuals with disabilities;

FIG. 10 is a perspective view of a mouse incorporating the reduced keyboard disambiguating system of the present invention;

FIG. 11 is a perspective view of a two-way pager incorporating the reduced keyboard disambiguating system of the present invention;

FIG. 12 is a front elevation of an alternate embodiment of a reduced keyboard disambiguating system in accordance with the present invention having a plurality of keys for display of a selection list;

FIG. 13 is a front perspective of yet another alternate embodiment of a reduced keyboard disambiguating system in accordance with the present invention, having a select key and a delete key positioned on either side of an array of data keys;

FIG. 14 is a front perspective of still another alternate embodiment of a disambiguating system in accordance with the present invention, having a control disk replacing the reduced keyboard;

FIGS. 15A–15C are perspective and cross-section views of still another embodiment of a reduced keyboard disambiguating system in accordance with the present invention having a pen-shaped body.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

I. System Construction and Basic Operation

With reference to FIG. 1A a reduced keyboard disambiguating system **50** formed in accordance with the present invention is depicted incorporated in a palmtop portable computer **52**. Portable computer **52** contains a reduced keyboard **54** and a display **53**. For purposes of this application, the term "keyboard" is defined broadly to include any input device having discrete mechanical keys, membrane keys, or defined areas on a touch screen, for example. Keyboard **54** has a reduced number of data entry keys from a standard QWERTY keyboard. Preferably, the keyboard contains twelve standard full-sized keys arranged in three columns and four rows. More specifically, the preferred keyboard contains nine data keys **56** arranged in a 3-by-3 array, and a bottom row of three system keys **58**, including a select key **60**, a shift key **62**, and a delete key **64**. A preferred arrangement of the letters and symbols on each key in the keyboard **54** is depicted in FIG. 1B.

Data is input into the disambiguation system via keystrokes on the reduced keyboard **54**. As a user enters a keystroke sequence using the keyboard, text is displayed on the computer display **53**. Two regions are defined on the display to display information to the user. An upper text region **66** displays the text entered by the user and serves as a buffer for text input and editing. A selection list region **70**, located below the text region, provides a list of words and other interpretations corresponding to the keystroke sequence entered by a user. As will be described in additional detail below, the selection list region aids the user in resolving the ambiguity in the entered keystrokes.

A block diagram of the reduced keyboard disambiguating system hardware is provided in FIG. 2. The keyboard **54** and the display **53** are coupled to a processor **100** through appropriate interfacing circuitry. A speaker **102** is also coupled to the processor. The processor **100** receives input from the keyboard, and manages all output to the display and speaker. Processor **100** is coupled to a memory **104**. The memory includes a combination of temporary storage media, such as random access memory (RAM), and permanent storage media, such as read-only memory (ROM), floppy disks, hard disks, or CD-ROMs. Memory **104** contains all software routines to govern system operation. Preferably, the memory contains an operating system **106**, disambiguating software **108**, and associated vocabulary modules **110** that are discussed in additional detail below. Optionally, the memory may contain one or more application programs **112**, **114**. Examples of application programs include word processors, software dictionaries, and foreign language translators. Speech synthesis software may also be provided as an application program, allowing the reduced keyboard disambiguating system to function as a communication aid.

Returning to FIGS. 1A and 1B, the reduced keyboard disambiguating system **50** allows a user to quickly enter text or other data using only a single hand. Data is entered using the data keys **56**. Each of the data keys has multiple meanings, represented on the top of the key by multiple letters, numbers, and other symbols. (For the purposes of this disclosure, each data key will be identified by the symbols in the center row of the data key, e.g., "RPQ" to identify the upper left data key.) Since individual keys have multiple meanings, keystroke sequences are ambiguous as to their meaning. As the user enters data, the various keystroke interpretations are therefore displayed in multiple regions on the display to aid the user in resolving any ambiguity. A selection list **76** of possible interpretations of the entered keystrokes is provided to the user in the selection list region **70**. The first entry **78** in the selection list is selected as a default interpretation and displayed in the text region **66** at an insertion point **88**.

The selection list **76** of the possible interpretations of the entered keystrokes may be ordered in a number of ways. In a normal mode of operation, the keystrokes are initially interpreted as the entry of letters to spell a word (hereinafter the "word interpretation"). Entries **78** and **80** in the selection list are therefore words that correspond to the entered keystroke sequence, with the entries ordered so that the most common word corresponding to the keystroke sequence is listed first. For example, as shown in FIG. 1A, a keystroke sequence ADF, OLX, NBZ and EWV has been entered by a user. As keys are entered, a vocabulary module look-up is simultaneously performed to locate words that have matching keystroke sequences. The words identified from the vocabulary module are displayed to the user in the selection list **76**. The words are sorted according to frequency of use, with the most commonly used word listed first. Using the example keystroke sequence, the words "done" and "doze" were identified from the vocabulary module as being the most probable words corresponding to the keystroke sequence. Of the two identified words, "done" is more frequently used than "doze," so it is listed first in the selection list. The first word is also taken as the default interpretation and provisionally posted as highlighted text at the insertion point **88**.

Following entry of the keystroke sequence corresponding to the desired word, the user presses the select key **60**. Pressing the select key draws a box around the first entry in the selection list **76** and redisplays the first entry at the insertion point **88** with a box around the entry. If the first entry in the selection list is the desired interpretation of the keystroke sequence, the user continues to enter the next word using the data keys **56**. The reduced keyboard disambiguating system interprets the start of the next word as an affirmation that the currently selected entry (in this case, the first entry in the selection list) is the desired entry. Alternatively, the selection of the first entry may occur after a user-programmable time delay. The default word therefore remains at the insertion point as the choice of the user, and is redisplayed in normal text without special formatting.

If the first entry in the selection list is not the desired interpretation of the keystroke sequence, the user may step through the items in the selection list by repeatedly pressing the select key **60**. For each press of the select key, the next entry in the selection list is boxed, and a copy of the entry provisionally copied to the insertion point and boxed. Provisionally posting the next entry to the text region allows the user to maintain their attention on the text region without having to refer to the selection list. If the second entry in the selection list is the desired word, the user proceeds to enter the next word after two presses of the select key and the disambiguating system automatically posts the second entry to the text region as normal text. If the second entry is not the desired word, the user may examine the selection list and press the select key a desired number of times to select the desired entry before proceeding to enter the next word. When the end of the selection list is reached, additional presses of the select key causes the selection list to scroll and new entries to be added to the end of the selection list. Those entries at the top of the selection list are removed from the list displayed to the user. The entry selected by multiple presses of the select key is automatically posted to the text region when the user presses any data key **56** to continue to enter text.

In the majority of text entry, keystroke sequences are intended by the user as letters forming a word. It will be appreciated, however, that the multiple characters and symbols on the keys allow the individual keystrokes and keystroke sequences to have several interpretations. In the preferred reduced keyboard disambiguating system, various different interpretations are automatically determined and displayed to the user at the same time as the keystroke sequence is interpreted and displayed to the user as a list of words.

For example, the keystroke sequence is interpreted as word stems representing all possible valid sequences of letters that a user may be entering (hereinafter the "stem interpretation"). Unlike word interpretations, word stems are incomplete words. When stem interpretations are displayed as part of the selection list **76**, the stem interpretations in the selection list are therefore not selectable by pressing the select key. By indicating the last keystrokes, however, the word stems allow the user to easily resume typing when his or her attention has been diverted in the middle of the word. As shown in FIG. 1A, the keystroke sequence ADF OLX NBZ EWV has been interpreted as forming a valid stem "albe" (leading to the word "albeit"). The stem interpretation is therefore provided as entry **81** in the selection list. Preferably, the stem interpretations are sorted according to the frequency of the most probable words that can be generated from each stem. When listing a stem interpretation in the selection list, the stem is omitted if a stem interpretation duplicates a word that is shown in the selection list. When the stem is omitted, however, the word corresponding to the omitted stem is marked with a symbol to show that there are also words of longer length having this word as their stem. Stem interpretations provide feedback to the user by confirming that the correct keystrokes have been entered to lead to the entry of a desired word.

Each pair of keystrokes is also interpreted as specifying a single character using a two-stroke specification method (hereinafter the "two-stroke interpretation"). The data keys **56** contain up to nine characters that are arranged in a 3-by-3 array on the top of each key. The first keystroke in each two-stroke pair of keystrokes is ambiguous—it tells the system that the user wishes to choose one of the nine characters grouped on the depressed key, but it does not specify which character.

The second keystroke qualifies or disambiguates the first. The position of the second keystroke in the 3-by-3 array of data keys specifies the character to be chosen from the 3-by-3 array of characters on the top of the first key. Each pair of keystrokes is therefore also interpreted by the reduced keyboard disambiguating system and automatically presented to the user in the selection list. For example, as shown in FIG. 1A, the entry of a keystroke sequence ADF and OLX first designates the top center data key, then the

character on that key in the left position of the second row, namely, the letter "a". The next two keystrokes NBZ and EWV designate the top right data key, then the symbol in the center position of the second row, namely, the letter "b". The two-stroke interpretation "ab" is therefore provided as an entry **82** in the selection list. It will be appreciated that the two-stroke interpretation may also be reversed, with the first keystroke qualifying or disambiguating the second.

A second method is also employed in which a sequence of keystrokes is interpreted as unambiguously specifying a specific string of alphabetic characters (hereinafter the "multiple-stroke interpretation"). The data keys **56** contain up to three letters that are arranged in a the center row of the 3-by-3 array of characters on the top of each key. The letters are ordered from left to right on each keytop in order of decreasing frequency. That is, the most frequently occurring letter appears first. Each letter of a word may be unambiguously spelled by pressing the key on which the letter appears a number of times corresponding to the position of the desired letter in the row. Thus, the leftmost letter is specified by a single keypress, the center letter by two keypresses, and the rightmost letter by three keypresses. When two successive letters in a word appear on the same key, the user must pause for a minimum delay period between the keypresses for the two letters. In general, this delay may be set quite short since the double and triple keypresses for letters can be expected to occur quite rapidly. The present invention, by virtue of the assignment of letters to minimize ambiguity and the ordering of letters on each key according to letter frequency, requires on average less than 69% of the keystrokes that would be required using the same method on a standard Touch-Tone keypad. The multiple-stroke interpretation of each sequence of keystrokes is therefore also performed by the reduced keyboard disambiguating system and automatically presented to the user in the selection list. For example, as shown in FIG. 1A, the entry of the keystroke sequence ADF, OLX, NBZ, EWV first designates the character string "aone". The multiple-stroke interpretation "aone" is therefore provided as an **35** entry **83** in the selection list. It will be appreciated that a given method for interpreting keystroke sequences may be easily enabled or disabled. For simplicity, the multiple-stroke interpretation is shown only in FIG. 1A, and is not shown in the other examples.

The keystroke sequence is also interpreted as a string of numerical digits (hereinafter the "numeric interpretation"). Data keys **56** contain characters representing numerical digits. One of the interpretations provided in the selection list is therefore the numerical digits that correspond to the keystroke sequence. For example, entry **84** is the numeric interpretation ("8495") of the keystroke sequence ADF, OLX, NBZ, EWV.

Finally, any keystroke sequence may be given additional meanings by linking the keystroke sequence to an object in a vocabulary module (discussed below). For example, as shown in the selection list in FIG. 1A, the keystroke sequence may be interpreted and presented as an entry **86** that corresponds to a system command or system menu. The system command "<cancel>" corresponds to a system macro object that cancels the current key sequence. Entry **86** may also correspond to a system menu. Selecting an entry labeled "<delete>", for example, may cause a number of menu items such as "delete file" or "delete paragraph" to be displayed in the selection list. The user would select the appropriate menu item by pressing the select key to box the desired item. Those skilled in the art will recognize that other system commands or system menus may also be defined in the system.

As noted above, in the normal mode of operation the entries in the selection list **76** corresponding to words are presented first in the list. In other circumstances, it may be desirable to have other keystroke sequence interpretations presented first in the list. For example, in situations where a series of numbers are to be entered, it would be desirable to have the numeric interpretation of the keystroke sequence presented first. The reduced keyboard disambiguating system therefore allows a user to select between other modes of operation by accessing a system menu. In a numeric mode of operation, the first interpretation provided in the selection list is the number corresponding to the keystroke sequence. In a two-stroke specification mode, the two-stroke interpretation is provided first in the selection list. The two-stroke specification mode therefore allows the user to enter a large number of words that must be spelled because they are not contained in the system vocabulary modules. Each of these modes of operation changes the ordering of the selection list displayed to the user.

The operation of the reduced keyboard disambiguating system is governed by the disambiguation software **108**. FIG. **3** is a flow chart of a main routine of the disambiguation software that generates a selection list to aid the user in disambiguating ambiguous keystroke sequences. At a block **150**, the system waits to receive a keystroke from the keyboard **54**. At a decision block **152**, a test is made to determine if the received keystroke is the select key. If the keystroke is not the select key, at a block **154** the keystroke is added to a stored keystroke sequence.

At a block **156**, objects corresponding to the keystroke sequence are identified from the vocabulary modules in the system. Vocabulary modules are libraries of objects that are associated with keystroke sequences. An object is any piece of stored data that is to be retrieved based on the received keystroke sequence. For example, objects within the vocabulary modules may include numbers, letters, words, stems, phrases, or system macros. Each of these objects is briefly described in the table below:

Object	Corresponding data
Numbers	A number, each digit of which corresponds to a single keystroke, e.g., the four-digit sequence "8495".
Letters	A letter or sequence of letters corresponding to sequences of keystrokes, e.g., the two letter sequence "ab". Each sequence of keystrokes is disambiguated using the two-stroke specification method of inputting individual letters and using the multiple-stroke specification method.
Word	A word corresponding to single or multiple keystrokes, e.g., the four letter word "done".
Stem	A sequence of letters representing a valid portion of a longer sequence of letters forming a word, e.g., "albe" as a stem of the word "albeit."
Phrase	A user-defined or system-defined phrase corresponding to single or multiple keystrokes, e.g., "To Whom it May Concern:".
System Macro	A word and associated code describing a system or user-defined function, e.g., "<clear>" to perform the function of clearing the current text region. In addition to the descriptive word, in the vocabulary module the system macro object is associated with the executable code necessary for performing the specified function.

While the preferred vocabulary objects are discussed above, it will be appreciated that other objects may be contemplated. For example, a graphic object may be associated with a stored graphic image, or a speech object may be associated with a stored segment of speech. A spelling

object may also be envisioned that would link the keystroke sequence of commonly misspelled words and typing errors with the correct spelling of the word. For example, words that include the letter sequence "ie" or "ei" will appear in the list of words even if the keystrokes for these letters are reversed from their proper sequence. To simplify processing, each vocabulary module preferably contains similar objects. It will be appreciated, however, that various objects may be mixed within a vocabulary module.

A representative diagram of a vocabulary module **110** is depicted in FIG. **4**A. A tree data structure is used to organize the objects in a vocabulary module based on a corresponding keystroke sequence. As shown in FIG. **4**A, each node N_1, N_2, . . . N_9 in the vocabulary module tree represents a particular keystroke sequence. The nodes in the tree are connected by paths P_1, P_2, . . . P_9. Since there are nine ambiguous data keys in the preferred embodiment of the disambiguating system, each parent node in the vocabulary module tree may be connected with nine children nodes. Nodes connected by paths indicate valid keystroke sequences, while the lack of a path from a node indicates an invalid keystroke sequence.

The vocabulary module tree is traversed based on a received keystroke sequence. For example, pressing the first data key from the start block traverses path P_1 to node N_1. Pressing the ninth data key after pressing the first data key traverses path P_9 to node N_9. As will be described in greater detail below, each node is associated with a number of objects corresponding to the keystroke sequence. As each node is reached, an object list is generated of the objects corresponding to the keystroke sequence. The object list from each vocabulary module is used by the main routine of the disambiguating system to generate a selection list **76**.

FIG. **4**B is a block diagram of a preferred data structure **400** associated with each node. The data structure contains information that links each parent node to children nodes in the vocabulary module tree. The data structure also contains information to identify the objects associated with the particular keystroke sequence represented by the node.

The first field in the node data structure **400** is a pointer bits field **402** that indicates the number and identity of children nodes that are connected to the parent node. Since there are nine data keys, only nine children nodes may be connected to any parent node. In the preferred embodiment, nine pointer bits are therefore provided in the pointer bits field to indicate the presence of a child node. Each pointer bit is associated with a pointer field **404***a*, **404***b*, . . . **404***n* that contains a pointer to the respective child node data structure in the vocabulary module. Since a child node is only present if the keystroke associated with the child node forms part of a valid keystroke sequence with the keystroke sequence associated with the parent node, the number of pointer fields varies for each node. For example, pointer bits field **402** may indicate that only six of the possible nine keystrokes lead to a valid child node. Because there are only six valid paths, only six pointer fields **404***a*, **404***b*, . . . **404***f* are included in the data structure for the parent node. The pointer bits field **402** is used to ascertain the identity of the pointer fields contained within the node data structure. If a keystroke does not lead to a valid child node, the associated pointer field may be omitted from the node data structure in order to conserve the amount of memory space required to store the vocabulary module.

Associated with each node are a number of objects that correspond to the keystroke sequence represented by the node. For each node, a number of objects field **406** is provided to indicate the number of objects (NUMOBJ) associated with the node. Since each node is associated with one and only one keystroke sequence, the number of objects associated with any given node is a constant. Each of the objects is associated by an object packet **408** contained in the node data structure. The number of objects field **406** specifies the number of object packets **408** that are present in the node data structure.

Each object packet **408** describes one object corresponding to the keystroke sequence represented by each node. Describing an object requires maintaining two object lists. FIG. **4**C depicts representative object lists created for a parent and a child in a vocabulary module tree. Object list **430** is an object list containing objects OL(**1**)–OL(**8**) associated with a node representing two keystrokes. Object list **440** is an object list containing objects NOL(**1**)–NOL(**8**) associated with a node representing three keystrokes. Each object list contains a list of all objects that are associated with each node. Object list **430** is associated with a parent node representing the keystroke sequence ADF OLX. Object list **440** is associated with a child node representing the keystroke sequence ADF OLX EWV. Although a maximum of eight entries are depicted as capable of being stored in each object list, it will be appreciated that the size of the object list may be varied to account for the maximum number of objects associated with each node.

Each object associated with a child node is constructed by adding a character sequence onto an object that was constructed for the parent node. The object packet **408** therefore contains a previous object identifier field **410** that identifies from a parent node object list an object that is used to construct the child node object. For example, with reference to FIG. **4**C, the third object "fo" in the old object list **430** is used to construct the first object "foe" in the new object list **440**. The previous object identifier field **410** therefore provides a link to the entries in the old object list to identify the old object used to construct the new object.

The object packet **408** contains a two-bit symbol field **412** to indicate the symbol to add to the identified object in order to construct the new object. In the preferred embodiment, each ambiguous key contains a maximum of three letters. The symbol field bits therefore specify the letter from each key that is used to construct the new object using the following binary code: "00" corresponds to the first letter on the key, "01" corresponds to the second letter on the key, and "10" corresponds to the third letter on the key. For example, with reference to FIG. **4**C, the first object "FOE" in the new object list **440** is constructed by using the third object "FO" in the old object list **430** and adding an additional keystroke to specify the E. In the preferred keyboard arrangement, "E" is the first letter on the EWV key, therefore the symbol field corresponding to the object "FOE" is set to "00" to indicate the first letter on the key. Encoding the objects in this manner greatly reduces the amount of storage space required for each vocabulary module. The encoding technique also allows direct access to vocabulary module entries without searching. Rather than having to store every object in the vocabulary module, a new object is defined using the two-bit code to add onto an old interpretation. The disclosed storage method requires, however, maintaining an object list from a parent in the vocabulary module tree in order to construct an object list of the child.

Symbol field **412** may also be set to the value "11". When set to the value "11", the symbol field indicates the presence of an ASCII sequence field **414** immediately following the symbol field. The ASCII sequence field is used to store strings of characters that are to be appended to the identified object. For example, the ASCII sequence field may store the

string "rward" to be added to the third object "fo" from the old object list to form the word "forward". In this manner, the length of an entered keystroke sequence does not necessarily directly correspond to the length of an associated object. The ASCII sequence field allows a vocabulary object to be identified by an arbitrary key sequence, i.e., stored at an arbitrary location within the vocabulary module tree.

The capability of storing objects with an arbitrary keystroke sequence is used to speed system processing of abbreviations and contractions. Abbreviations and contractions are typically identified by a keystroke sequence that corresponds to their pure alphabetic content, ignoring punctuation. The result is that abbreviations and contractions are easily accessed by the user without entering punctuation, resulting in a significant savings in keystrokes. For example, the user can enter the keystroke sequence for "didnt" without adding an apostrophe between the "n" and the "t". The word in the vocabulary module that corresponds to the keystroke sequence "didnt" contains an ASCII sequence field with an apostrophe between the "n" and the "t". The disambiguating system will therefore automatically display to the user the correct word "didn't", without requiring the user to enter the punctuation mark. The disambiguating system uses the same technique to properly display foreign words having unique characters (such as "Ü", which may be entered as a "U"). Capitalization may be handled in a similar manner. Words that should always be used in all capital letters, with an initial capital letter, or with a capital letter in the middle are identified by keystroke sequences without keystrokes indicating capitals, eliminating the need for the user to enter such capitalization.

An object type field **416** may also be included in each object packet **408** to specify additional information about the object being constructed. The object type field may contain a code to specify whether the generated object is a word, a word stem, or any other object. The object type field therefore allows different types of objects to be mixed within a given vocabulary module. Moreover, the object type field may also include information regarding the part of speech of the word, information about how the object is capitalized, or information needed to construct various inflections and endings. A reduced keyboard disambiguating system using a vocabulary module having the part of speech information may use the additional information to implement syntactical analysis to improve the disambiguation process. The object type field may also contain a unique code to allow transmission of text in a compressed form. The unique code would be transmitted to a remote terminal instead of transmitting the entered keystroke sequence or the associated disambiguated characters.

One of the key features of the preferred vocabulary module tree data structure is that the objects associated with each node are stored in the node data structure **400** according to their frequency of use. That is, the first object packet **408** has a higher frequency of use than the second object packet in the node data structure, which has a higher frequency of use than the third object packet. In this manner, the objects are automatically placed in the object list so that they are sorted according to decreasing frequency of use. For purposes of this description, frequency of use refers to the likelihood of using a given word within a representative corpus of use, which is proportional to the number of times that each word occurs in the corpus.

While preferably the objects are stored within the node data structure **400** in order according to their frequency of use, it will be appreciated that a frequency of use field could also be associated with each object packet. The frequency of use field would contain a representative number that corresponds with the frequency of use of the associated object. The frequency of use between different objects would be determined by comparing the frequency of use field of each object. The advantage of using the latter construction that associates a frequency of use field with each object packet is that the frequency of use field could be changed by the disambiguating system. For example, the system could change a frequency of use field to reflect the frequency with which a user used certain objects within the vocabulary module during representative text entry.

Returning to FIG. **3**, at block **156** those objects that correspond to the received keystroke sequence are identified in each vocabulary module. FIG. **5** is a flow chart of a subroutine **500** for analyzing the received keystroke sequence to identify corresponding objects in a particular vocabulary module. The subroutine **500** constructs an object list for a node representing a particular keystroke sequence. As noted above, to construct a new object list the disambiguating system starts with a copy of the old object list. At a block **502**, the object list from the prior node is therefore stored so that it may be used to construct the new object list.

In the main routine shown in FIG. **3**, a keystroke was detected by the system at block **150**. The receipt of a new keystroke causes a downward traversal in the vocabulary module tree, if a valid path exists to a child corresponding to the keystroke. At a block **504** in FIG. **5**, the pointer bits field of the parent node data structure is therefore examined to determine if a pointer corresponds to the received keystroke. At a decision block **506**, a test is made of the pointer bits field to determine if a pointer field **404***a*, **404***b*, . . . **404***n* exists that corresponds to the entered keystroke. If no pointer field corresponds to the keystroke, at a block **508** the old object list is copied to the new object list. At a block **510**, the object list is returned to the main routine to generate the selection list. Since the received keystroke is part of an invalid keystroke sequence that does not correspond to any object within the vocabulary module, the keystroke is ignored and the current object list is returned to the main routine as being the object list from the vocabulary module. The branch of the subroutine **500** that comprises blocks **508** and **510** therefore ignores any invalid keystroke sequences and returns the object list generated at the parent node for possible inclusion in the selection list generated by the disambiguating system.

If a pointer exists corresponding to the received keystroke at decision block **506**, the subroutine proceeds to a block **512** where the pointer is followed to the child node representing the keystroke. When the child node is identified, a new object list corresponding to the node must be constructed. At a block **514**, on identifying the child node, the number of objects associated with the node are determined from the number of objects field **406** in the child node data structure.

After determining the number of objects to be generated at the child node, the subroutine enters the loop comprised of blocks **516** through **526** to reconstruct the object list associated with the child node. At a block **516**, a counter is initially set to one. At a block **518**, a test is made to determine if the counter has exceeded the number of objects associated with the node. If the counter has not exceeded the number of objects associated with the node, at a block **520** the previous object identifier field **410** is examined and the corresponding object loaded from the old object list. At a block **522**, the symbol field **412** is examined and the appropriate symbol associated with the received keystroke appended to the end of the identified object. It will be appreciated that an additional ASCII sequence may also be

appended to the identified object at block **522** if the symbol field indicates the presence of an ASCII sequence field **414** in the node data structure. At a block **524**, the combined object and symbol are stored as a new object in the new object list. After storing the new object in the object list, at a block **526** the counter is incremented by one. The subroutine then loops to decision block **518** to determine whether all of the objects associated with the node have been constructed.

If the test at decision block **518** indicates that all of the objects have been constructed for the node, the subroutine proceeds to a block **528** where the new object list is returned to the main routine in order to generate the selection list. It will be appreciated that the subroutine **500** for generating the object list associated with each node is performed for each keystroke received from the user. No "searching" of the vocabulary modules is performed as the user enters a new keystroke sequence, since each keystroke merely advances the subroutine one additional level within the vocabulary module tree. Since a search is not performed for each keystroke, the vocabulary module returns the list of objects associated with each node in a minimal period of time.

It will be appreciated that the relationship between vocabulary module objects and keystroke sequences is an implementation detail of the vocabulary module. If only a limited number of objects (i.e., fewer than a predetermined number) are associated with a particular node, additional nodes may be traversed to identify objects having a keystroke sequence starting with the entered keystroke sequence. The objects are identified by traversing downward in the vocabulary module tree along valid paths until the objects are identified. The objects are then placed in the selection list before all the keystrokes corresponding to the objects are entered. The objects are included in addition to the objects that are directly associated with the input keystroke sequence. Displaying objects associated with longer keystroke sequences in the selection list (hereinafter referred to as the "look-ahead" feature) allows the user to optionally select the objects immediately, without having to complete the remaining keystrokes to specify the object. The look-ahead feature is enabled when the number of objects identified in the vocabulary modules fails to fill the selection list region **70** on the display.

Returning to FIG. **3**, at blocks **158–162** the objects returned from the search of the vocabulary modules are prioritized and displayed to the user in the selection list **76**. To determine the sequence of objects displayed in the selection list, priorities are established between each vocabulary module and also between the returned objects from each vocabulary module.

To prioritize the object lists identified from the various vocabulary modules, at block **158** the mode of operation of the reduced keyboard disambiguating system is examined. As discussed above, in a normal mode of operation the word interpretations are displayed first in the selection list. The object list from a word vocabulary module would therefore be assigned a higher priority than the object list from the other vocabulary modules. Conversely, if the disambiguating system is in the numeric mode of operation, the numeric interpretations would be assigned a higher priority than the other vocabulary modules. The mode of the disambiguating system therefore dictates the priority between vocabulary module object lists. It will be appreciated that in certain modes, the object lists from vocabulary modules may be omitted from the selection list entirely.

Object lists generated from vocabulary modules may contain only a single entry, or they may contain multiple entries. At block **160**, the priority between the objects from the same vocabulary module is therefore resolved if the object list contains multiple entries. Within the search results from each vocabulary module, the objects that match a particular keystroke sequence are also given a priority that determines their relative presentation with respect to each other. As noted above, preferably the default presentation order is by decreasing frequency of use in a representative corpus of usage. The priority data associated with each object is therefore used to order the objects located in the search in the selection list. Since the selection list region **70** is limited in the number of entries that may be displayed, objects located by the search that fall below a predetermined minimum frequency of use may be omitted from the initial display of the selection list. The omitted objects may be later added to the selection list when the user scrolls beyond the end of the displayed list.

Many of the properties associated with the presentation of the vocabulary module search results are user-programmable by accessing appropriate system menus. For example, the user can specify the order of individual objects or classes of objects in the selection list region. The user may also set the priority level that determines the priority between vocabulary modules and between the objects identified from each vocabulary module. In this manner, the number of entries presented to the user in the selection list region may be kept to a minimum. Additional entries in the selection list region may always be scrolled into view by repeated presses of the select key.

After the priorities between the objects have been resolved, at a block **162** a selection list is constructed from the identified objects and presented to the user. As a default interpretation of the ambiguous keystroke sequence entered by the user, the first entry in the selection list is provisionally posted and highlighted at the insertion point **88** in the text region **66**. The disambiguating software routine then returns to block **150** to wait for the next keystroke.

If the detected keystroke is a select key, the "yes" branch is taken from decision block **152** to a block **164**. At a block **164**, a box is placed around the first entry in the selection list, and at the insertion point where it has been provisionally posted. At a block **165**, the system then waits to detect the next keystroke entered by the user. At a decision block **166**, a test is made to determine if the next keystroke is the select key. If the next keystroke is the select key, at a block **168** a box is placed around the next entry in the selection list and the entry is provisionally displayed at the insertion point with a box around the entry. The routine then returns to block **164** to detect the next keystroke entered by the user. It will be appreciated that the loop formed by blocks **164–168** allows the user to select various interpretations of the entered ambiguous keystroke sequence having a lesser frequency of use by depressing the select key multiple times.

If the next keystroke is not the select key, from decision block **166** the routine continues to a block **170** where the provisionally displayed entry is selected as the keystroke sequence interpretation and is converted to normal text formatting in the text region. At a block **172**, a space is added following the selected interpretation, since the receipt of an ambiguous keystroke following the select key indicates to the system the start of a new ambiguous sequence. At a block **174**, the old keystroke sequence is cleared from the system memory. The newly received keystroke is then used to start the new keystroke sequence at block **154**. Because the word interpretation having the highest frequency of use is always presented as the default choice, the main routine of the disambiguation software allows a user to continuously enter

text with a minimum number of instances where additional activations of the select key are required.

Ambiguities are further reduced in the reduced keyboard disambiguating system by assigning the letters on the keys to minimize the number of ambiguities that occur during text entry. For example, with the keyboard letter arrangement depicted in FIG. **1**, the desired word appears at the top of the selection list more than 99% of the time. For simple text, the reduced keyboard disambiguating system therefore typically requires less than one additional keystroke per hundred words typed. Such results are a great improvement over prior art methods of resolving reduced keyboard ambiguity, and make the reduced keyboard disambiguating system suitable for use in many text-entry applications.

II. Advanced System Features

1. Visual and Auditory Feedback

In FIG. 1A, information about the state of the reduced keyboard disambiguating system **50** is provided to the user using various types of visual and auditory feedback. Entries in the selection list **76** are presented in different colors depending on the interpretation of the keystroke sequence. For example, the word interpretation may be one color, the two-stroke interpretation another color, and the numeric interpretation a third color. A user may therefore easily scan the selection list for the desired interpretation.

During system use, the disambiguating system also indicates to the user which data keys **56** in the keyboard may constitute the next key in a valid word interpretation. As the number of keystrokes entered in a given keystroke sequence increases, the number of data keys that may be entered as the next key to form a word contained in the system vocabularies begins to decrease. Two techniques can be used to show the user which data keys may validly be used as the next keystroke when forming a word. First, a visual indication may be provided on the data keys themselves. If the data keys are typewriter-style keys, light-emitting diodes (LEDs) may be incorporated on the tops of each data key. The LEDs are lit when the data key is available for entry as the next keystroke to form a viable word interpretation. Multiple LEDs may also be incorporated on each key to indicate the specific letters that are valid on each key. Second, a separate display may be provided to the user that corresponds to the shape of the keyboard. For example, a three-by-three grid of LEDs may be presented to the user adjacent to the keyboard, each LED corresponding to one of the data keys. Alternatively, a 3×3 grid may be displayed in the text region **66**, with elements in the grid turned on or off to indicate valid data keys to form a word interpretation.

It will be appreciated that providing a visual indication to the user of which data keys may be entered to form a valid word does not prevent the user from using the other data keys on the keyboard. Especially when entering words or special punctuation that are not contained within the disambiguating system vocabularies, the user must be able to use all possible keys in order to enter a word using the two-stroke method.

Audible tones indicate the state of selection list **76** and provide feedback about keystrokes to allow system information to be conveyed independently of any visual feedback provided in the selection list. Distinct tones indicate when the selection list is empty, when it contains a single unique word, and when it contains multiple ambiguous words. Another tone indicates when the second or third entry in the selection list has a frequency of use above a preset threshold, or when the difference in frequency between the first and second word falls below a selected threshold. Still other tones distinguish the type of item being selected in the selection list as the select key is pressed. Separate tones are therefore used to distinguish words, numbers, proper nouns, phrases, system macros, etc. Distinct tones can also be assigned to each key to allow identification of mistakes in keystrokes. Finally, a unique tone is heard when the user presses a key that is unavailable for a word as described above.

Additional auditory feedback may be provided to the user by including a voice synthesizer as an application program **112, 114** in the disambiguating system. As a user enters keystrokes, the voice synthesizer announces the first entry in the selection list. To allow typing to proceed unimpeded, the first entry is announced after a slight delay. The user may also cause the first entry to be spoken immediately by pressing the select key. The auditory feedback provided by a voice synthesizer allows visually-impaired users to use the system without having to view the selection list.

2. Keying Techniques

The reduced keyboard disambiguating system **50** supports four keying techniques to perform system commands, access system menus, or enter special symbols including punctuation and diacritics. First, the disambiguating system supports latching keystrokes, performed by pressing and holding a key in the depressed position for a specified minimum time. The minimum time a key must be held before being latched is set by the user in a system menu. When the minimum time has elapsed, audible or visual feedback is provided to the user. For example, a clicking noise may be made when a key is depressed for a sufficient time for the key to latch. The latching of a key may be combined with a second keystroke and mapped to any disambiguating system command, menu, or symbol. For example, a short press of the delete key **64** causes the last keystroke to be deleted, while latching the delete key causes the disambiguating system to remap the data keys **56** so that the next data key keystroke implements a system delete command. One data key may be remapped to act as a "delete line," another data key as a "delete paragraph," and a third data key as a "delete document," and so on. After performing the system command, the data keys return to their unmapped states.

Second, the disambiguating system supports overlapping keystrokes, performed by pressing and holding a first key while a second key is pressed. For example, the user could press and hold the delete key **64** before pressing the ADF key in the data keys **56**. A minimum delay is required between depressing and holding the first key and depressing and holding the second key to distinguish the overlapping keystroke from a chorded keystroke discussed below. The various overlapping keystroke combinations may be mapped to any disambiguating system command, menu, or symbol.

Third, the disambiguating system supports chorded keystrokes, performed by pressing two or more keys at approximately the same time and holding them for at least a minimum delay time. For example, the user could simultaneously press and hold the delete key **64** and the CYK key in the data keys **56**. The two keys are considered to have been pressed simultaneously if the interval between the two key presses is shorter than a maximum delay time. Both the minimum delay time and the maximum delay time of the chorded keystrokes may be set by a user in a system menu. The various chorded keystroke combinations may be mapped to any disambiguating system command, menu, or symbol.

It will be appreciated that certain overlapping keystrokes or chorded keystrokes are physiologically easier to implement than others. For example, it is easier to enter chorded

keystrokes using two keys in the same row. Overlapping and chorded keystrokes that are easier to implement are therefore assigned to more frequently used commands, menus, or symbols. It will also be appreciated that the unambiguous row of system keys **54** may also be used in latching, overlapped, or chorded combinations.

Fourth, the disambiguating system supports the mapping of single keys to any disambiguating system command, menu, or symbol. The English language only contains two one-letter words ("A" and "I") that must be presented as the first choice in the selection list when the respective single keys are pressed. Pressing any of the other data keys **56** that do not contain "A" or "I" can therefore be used to list a system command, a menu, or a symbol as the first item in the selection list.

It will be appreciated that while the four keying techniques discussed above are preferably implemented in the reduced keyboard disambiguating system, depending on the keyboard construction some of the keying techniques may not be available. For example, the use of certain types of keyboards to implement keyboard **54**, such as touch screens or thin-film keyboards, may make it impossible to detect chorded, latching, or overlapped keystrokes.

3. Punctuating, Capitalizing, Spacing, and Executing

When entering text, the reduced keyboard disambiguating system **54** allows punctuation to be entered using several techniques. As shown in FIG. 1B, common punctuation marks are provided on the tops of the data keys **56**. A user may therefore use the two-stroke method to specify when a punctuation mark is to be included in the text. Alternatively, a user may use the latching, overlapped, or mapping of single key methods discussed above to enter a punctuation mark. Other punctuation marks and symbols that are used less often are assigned a chorded keystroke sequence or are accessible from a system menu. When entering a letter having a diacritic, an appropriate keystroke indicating the type of diacritic mark is entered in the keystroke sequence either immediately before the letter containing the diacritic.

Many common words contain a punctuation mark, however, such as a hyphen, apostrophe, or diacritic. As discussed above in the description of the vocabulary modules, the user may enter the keystroke sequence that corresponds only to the letters of the word without regard to a punctuation mark during entry of common words containing a punctuation mark. All techniques for entering punctuation are simultaneously enabled during system operation.

The shift key **62** preferably operates in the following manner. Pressing the shift key causes the next data keystroke to be capitalized. Pressing the shift key multiple times cycles through various functions. A second shift keystroke acts as a "ctrl" key, a third shift keystroke operates as an "alt" key, and a fourth keystroke operates as a "caps lock" key. A fifth shift keystroke cancels the operation of the shift key, returning to an unshifted state.

When the first letter of the word is capitalized, the words in the selection list **76** are optionally ordered to show proper nouns listed first in the list of interpretations. The words are sorted according to frequency of use, with the most commonly used proper nouns listed first. The frequency of occurrence of the proper nouns may be prestored in a vocabulary module, programmed by the user via a system menu, or adaptively calculated as the user continues to use the system as discussed below. The display of proper nouns first in the selection list is disabled or a warning is sounded when the capitalized word is the first word in a sentence.

4. Editing

A user of the reduced keyboard disambiguating system **50** may enter an editing mode via an overlapped keystroke sequence. FIG. **6** depicts the display **53** of the computer while in editing mode. On entering the editing mode, a key map **800** indicating the remapped data key functions is displayed to the user in the text region **66**. Each box in the key map indicates the editing command that will result if the corresponding data key is pressed.

Editing is performed on a word-by-word basis. Once in the edit mode, the user selects a word to edit by moving a cursor **802** to highlight the word. The user selects a word to edit by using the data keys corresponding to the commands "←word" and "word→" to move through the text displayed in the text region. Selecting a word in the text region recreates the same selection list **76** that was presented to the user at the time the edited word was added to the text region **66**. To recreate the original selection list, the most efficient key sequence that would produce a given word is reconstructed from the word itself The keystroke sequence is then interpreted using the vocabulary modules to reconstruct the environment (i.e., the contents of the selection list) from which the original word was selected.

Once a word is selected and the selection list redisplayed, the user has several different options for editing the word. One option allows the user to select a different word from the displayed selection list by appropriately pressing the select key. Multiple presses of the select key move down the selection list, in the same manner in which the edited word was originally selected. Pressing the delete key moves up the interpretation list. The boxed word from the selection list is automatically added to the text region when the user leaves the editing mode. After any word is selected in editing mode, additional text typed is inserted after the word at the point being edited in the text region.

Editing of a particular word is completed by selection of an appropriate word from the selection list. When the editing of a word is complete, the user can use the "←word" and "word→" keys to move through the text string displayed in the text region and select a different word for editing. When editing of the text is complete, the user exits the editing mode using the "exit" key.

5. "Respell" Mode

A second editing option is a "respell" mode, which allows the user to disambiguate an ambiguous keystroke sequence on a letter by letter basis. The respell mode is used when a user types a word that is not in the system vocabulary using only a single keystroke for each letter. Finding no words in the vocabulary module matching the keystrokes, the user may then enter the respell mode using an overlapped keystroke, and disambiguate each letter. The respell method of entering an unknown word in this manner is similar to the two-stroke method except the disambiguating process follows entry of the entire word, rather than after each letter of the word.

On receipt of the overlapped keystroke that designates respell mode, the preceding sequence of ambiguous keystrokes is displayed as a series of key icons, each key icon depicting the originally entered keystroke. A representative display is shown in FIG. **7**, with four key icons **810** appearing for the original key sequence ADF, OLX, NBZ, EWV. The first key icon in the word is highlighted in the text region **66**. For each key icon, the user presses a disambiguating key whose position in the middle row of data keys **56** corresponds to the position of the desired letter in the highlighted icon. The key used to disambiguate the keystroke is the same key that is used as the second key when entering letters using the two-stroke method. On receipt of each disambiguating keystroke, the highlighted icon is replaced by the selected letter, and the next icon in the

sequence is highlighted. The disambiguated character sequence also appears in the selection list. Respell mode is terminated when the last keystroke icon is disambiguated, when the select key is pressed, or by the same overlapped keystroke that initiates respell mode.

6. Shortcuts

The reduced keyboard disambiguating system **50** incorporates several shortcuts that speed entry of text into the system. One shortcut causes the entry of an unambiguous symbol or an unambiguous function to delimit a current ambiguous sequence, if there is one, and automatically select and accept the first entry in the selection list **76**. For example, a user that enters the ambiguous sequence CYK ADF NBZ, followed by an explicit (e.g., by overlapped keystroke) apostrophe ('), will have the system automatically select and post to the text region the word "can'", since "can" is the first (i.e. most likely) interpretation of the key sequence CYK ADF NBZ. When a keystroke sequence is automatically interpreted in this manner, no space is generated following the selected interpretation. The user may therefore continue to enter the word by adding additional characters. This shortcut is typically used when punctuation marks are used in a keystroke sequence.

In addition to operating in different modes of operation wherein the selection list **76** is ordered to present a selected type of keystroke interpretation as the first entry in the list, the reduced keyboard disambiguating system **50** also may enter via a system menu a number of dedicated modes wherein only one interpretation is made for each key and only one or no entries are displayed in the selection list. For example, in a dedicated numeric mode, each keystroke corresponds to the entry of a number. In a dedicated cursor movement mode, each of the outside circle of data keys corresponds to a cursor movement direction to allow a user to manipulate a cursor in an application program. Those skilled in the art will recognize that other dedicated operating modes may also be envisioned, such as a mouse emulation or Touch-Tone phone emulation. When operating in the dedicated modes, text or commands are directly implemented since there is no ambiguity in the entered keystrokes.

Another shortcut provided by the disambiguating system allows unambiguous keystroke to immediately select certain interpretations in the selection list **76**. Preferably, if more than one entry of a particular interpretation is present, the entry in the selection list having the highest frequency of use is selected. The use of an unambiguous keystroke to select a desired interpretation eliminates having to step through an unpredictable number of other interpretations.

7. System Output

The disambiguated output from the reduced keyboard disambiguating system **50** system is generally provided to other application programs **112, 114** running on and sharing the resources of the disambiguating system. Text is therefore directly entered into the application program using the aid of a selection list, such as shown in the system of FIG. **1A**.

In other instances, the target for output is an application program running on a separate platform. For example, a user may desire to transfer entered text to a remote terminal. Those skilled in the art will recognize that a PCMCIA card or modem card may be added to computer **52** to allow data transfer with other devices. Disambiguated text may be transferred after the entry of each word, or after an explicit "send" function accessed by the user via a system menu.

8. Custom Vocabularies

Among the vocabulary modules **110** contained in the reduced keyboard disambiguating system **50** is a custom vocabulary module. Words entered using the two-stroke or multiple-stroke methods are automatically stored by the disambiguating system in the custom vocabulary module. The words stored in the custom vocabulary module will thereafter be automatically displayed in the selection list when the user enters the shorter sequence of single (ambiguous) keys for these words.

In addition to adding words to the custom vocabulary module during normal text entry, words may also be added to a user's custom vocabulary module from a variety of other sources. For example, documents may be downloaded into the disambiguating system and parsed to identify proper nouns or other words that are not contained in the vocabulary modules present in the disambiguating system. After parsing, the newly identified proper nouns and words are added to the user's custom vocabulary module. Custom vocabulary modules may also be uploaded or downloaded to other disambiguating systems or to a mass storage medium. A user may therefore merge their present custom vocabularies with other vocabularies created by another user.

The words in the selection list **76** identified from the standard vocabulary modules are preferably always presented to the user in the same order, according to decreasing frequency of use, so that the user can commit to memory the keystrokes necessary to enter a desired word.

III. Representative System Operation

FIGS. **8A** through **8J** depict the display **53** of the portable computer **52** during a representative use of the reduced keyboard disambiguating system. After turning on the power of the portable computer, the text region **66** and selection list region **70** are empty. In FIG. **8A** the user has pressed the ADF key. The word vocabulary module has interpreted the ADF key as the word "A" **902** and placed the interpretation, capitalized, in the selection list **76**. The stem vocabulary module has interpreted the ADF as the stems "D" and "F" **903**, and placed the interpretation in the selection list. The numeric vocabulary module has interpreted the keystroke as the number "8" **904**. The system command vocabulary module has matched the keystroke sequence with three system commands, "<Setup>" **906**, "<Edit>" **908**, and "<Cancel>" **910**, and added the system commands to the selection list. The first entry in the selection list has also been provisionally posted and highlighted in the text region at insertion point **88**.

In FIG. **8B** the user has pressed the select key **60**, selecting and boxing the word "A" **902** in the selection list. The copy of the word that had been provisionally posted at the insertion point **88** becomes boxed, rather than highlighted, to indicate that the interpretation will be the selected interpretation if the user continues to enter text with an ambiguous keystroke.

In FIG. **8C** the user has pressed the select key **60** four additional times, boxing the system command "<Setup>" **906** in the selection list. When the <Setup> command is boxed, a 3×3 key map **920** is displayed in the upper right corner of the text region **66**. Each square in the 3×3 key map directly corresponds with one of the data keys **56** in the keyboard. Pressing the center data key, for example, causes the reduced keyboard disambiguating system to perform the system command "Default Setup." A help dialog box **922** is also displayed in the text region to provide further instructions to a user.

In FIG. **8D**, the user has executed the "Load Vocab" command by depressing the upper left corner data key. Executing the command causes a system menu of vocabulary modules that can be loaded to be displayed in the selection list. The first entry in the selection list, the standard

system vocabulary module **924**, has been boxed and selected by default. It will be appreciated that the disambiguating system allows nesting of menus as depicted in FIGS. **8**C and **8**D That is, an entry in the selection list may remap the data keys to a number of system commands, while executing a system command may lead to further choices presented as a system menu in the selection list. Nesting the menus in this manner allows multiple system functions and commands to be implemented with a minimum number of keys.

FIG. **8**E results if the user presses the OLX key after FIG. **8**A. The word vocabulary module has interpreted the ADF and OLX keys as the words "Do" **930** and "Ax" **931**, and placed these words in the selection list **76**. The stem vocabulary module has interpreted the keystrokes as the stems "Fo" **932**, "Fl" **933**, "Ao" **934** and "Al" **935**. The two-stroke vocabulary module has interpreted the entered key sequence as the letter "A" **936**. The numeric vocabulary module has interpreted the key sequence as the number "84" **937**. The system command vocabulary module has matched the keystroke sequence with the system command "<cancel>" **938**, and added the command to the selection list **76**. The first entry from the selection list is provisionally posted at the insertion point **88** of the text region.

In FIG. **8**F, the user has subsequently pressed the NBZ key followed by the EWV key. The word vocabulary module has matched the keystroke sequence to two words, "Done" **940**, and "Doze" **941**. By default, multiple words are ordered in the selection list **76** so that the most frequently occurring word appears first. "Done" is therefore listed before "Doze" in the selection list. The interpretations from the other vocabulary modules, namely, "Albe" **942**, "Ab" **944**, "8495" **945**, and "<cancel>" **946** are provided in the selection list **76** after the word interpretation.

In FIG. **8**G, the user has pressed the IMG key. The word vocabulary module has found no words that exactly match the keystroke sequence ADF OLX NBZ EWV IMG. However, by "looking ahead" the word vocabulary module has located a longer word, "Albeit" **952**, and listed the word in the selection list as a possible interpretation. The two-stroke interpretation of the five keys results in the string "Ab?" **954** (the question mark indicating that the third letter is awaiting the sixth keystroke which will disambiguate the fifth), the numeric interpretation results in the number "84956" **956**, and the system command interpretation still corresponds to "<cancel>" **958**.

FIG. **8**H results from FIG. **8**F if the user presses the select key three times. The word stem "Albe" **942** is eliminated from the selection list on receipt of the first press of the select key. The two-stroke vocabulary module had added the string "Ab" **944** to the selection list. By pressing the select key three times, the user has highlighted the two-stroke interpretation, "Ab", causing a copy to be provisionally posted at the insertion point **88** in the text region **66**. When the user presses a data key to start the following word, the word "Ab" is selected as the keystroke interpretation. The disambiguating system also automatically adds the word "Ab" to the user's custom vocabulary module, allowing the user to subsequently spell "Ab" with the two keystroke sequence ADF NBZ.

FIG. **8**I results if the user presses and releases the select key after FIG. **8**G, selecting the word "Albeit" **952**. Pressing the select key causes the first entry in the selection list and the word at the insertion point to be boxed. The incomplete two-stroke specification "Ab?" **954**, which resulted from an odd number of keystrokes, is eliminated from the selection list when the select key is pressed.

If the user presses and holds the select key **60** after FIG. **8**G, instead of pressing and releasing the select key, then the display depicts the overlapped key meanings as shown in FIG. **8**J. The overlapped key meanings are displayed in a 3x3 key map **954**. Each box in the key map indicates the command that will result if the corresponding data key is pressed while the select key is being held down. A dialog box **956** is also provided to indicate that the overlapped keystroke combinations use the select key **60**. It will be appreciated that other system keys **58** could similarly be used to implement an overlapped keystroke. Chorded key combinations are accessible at any time and can be entered without disturbing other input processes.

IV. Alternate Applications and Embodiments

While the preferred embodiment of the invention has been illustrated and described, it will be appreciated that various changes can be made therein without departing from the spirit and scope of the invention. For example, those skilled in the art will appreciate that the keyboard **54** of the reduced keyboard disambiguating system may have as few as three or as many as twenty data keys. The disambiguation technique disclosed herein is equally applicable to keyboards of different sizes.

An embodiment of the reduced keyboard disambiguating system having eight keys is particularly suited for use by people with hand control disabilities. A suitable eight-key layout is shown in FIG. **9**. Seven keys are used to enter letters or numbers, and one key is used as the "select" key. The eight keys are organized in a circle with each key corresponding to one of eight directions. When organized in a circle, each key may be easily selected by movement of a joystick, head-pointing, or by movement of the eye as detected by an eye-tracking device. Having eight keys allows each key to be represented numerically by three bits. For individuals with disabilities who work best with a Morse-like code, each ambiguous keystroke can be encoded as Morse-like sequences with an average code length of two (i.e., the average number of "dots" and "dashes" per letter is only two). For individuals with disabilities who work best with scanning, the eight-key system requires scanning across only eight keys rather than 26 or more. With eight keys in the reduced keyboard disambiguating system, more than 97% of all words require no additional keystrokes. Even higher efficiencies can be achieved using the "look-ahead" feature of the vocabulary modules discussed above.

It will also be appreciated that the reduced keyboard disambiguating system of the present invention may be used in conjunction with any computer input device. For example, a disambiguating system of the present invention is depicted in FIG. **10** as incorporated in a mouse **1100**. The mouse is connected to a computer **1102** having a monitor **1104**. Text entry is performed in the manner described above using a reduced set of data keys **1106** located on the upper surface of the mouse. A button **1108** located on the side of the mouse operates as an unambiguous system key to delimit entered keystroke sequences and select different interpretations from the displayed selection list. Incorporating a keyboard directly on the mouse is particularly advantageous in that a user does not have to switch between a mouse and a keyboard when operating in a graphical computer interface.

The reduced keyboard disambiguating system may also be incorporated in any portable electronic device requiring accurate text entry using only a single hand. For example, in FIG. **11** a reduced keyboard disambiguating system of the present invention is depicted as being incorporated in a two-way pager **1200**. The two-way pager is provided with a touch-sensitive screen **1202** to display messages to a user and to allow the user to enter messages into the pager. Rather

than having a separate typewriter-style keyboard, a keyboard **1204** is displayed on the touch-sensitive screen. Contact by a user with the touchscreen at a location above a depicted key is interpreted by the system as a keystroke. To aid the user in locating the keys on the touchscreen, a transparent or semitransparent grid or raised area may be formed on the surface of the keypad above each key. The user would therefore receive a visual or tactile indication of the location of their fingers on the touchscreen with respect to the keyboard. It will be appreciated that the keyboard **1204** may also be overlaid over the area of the touchscreen that displays messages to the user. In this embodiment, the user would periodically move their hand to review the entered text on the display.

Text would be entered into the pager **1200** in a similar manner to the applications discussed above. Rather than displaying the selection list in a selection list region, however, a selection list **1206** in the pager application is displayed at the insertion point **88** in the text window. The first entry in the selection list is highlighted, and the user may select the first entry or other entries using the select key in the manner described above. Once the disambiguation of the keystroke sequence is complete, the selection list **1206** is removed and the selected entry left as the last word in the entered line of text. Displaying the selection list at the insertion point in the text speeds the text entry since the user's point of gaze does not have to be removed from the text being entered.

An alternate embodiment of a portable reduced keyboard disambiguating system **1300** is shown in FIG. **12**. The front panel of the system is a touch-sensitive, color liquid crystal display (LCD) **1302** that displays information to a user and also accepts user input. The entire front panel of the system is touch sensitive, so that contact with the surface of the reduced keyboard generates input signals to the system corresponding to the location of contact. The reduced keyboard contains twelve keys **1304** arranged in the preferred layout described above.

Rather than defining regions on a single display to display information to a user, several windows are provided in the disambiguating system **1300**. A text window **1306** displays several lines of text entered by the user to allow text input and editing. A keystroke window **1308** displays symbols corresponding to the individual keystrokes entered by the user in the order of entry. The keystroke window provides visual feedback to the user and also allows editing of the keystroke sequence.

Instead of a selection list, a set of selection list keys **1310** are provided to display a selection list corresponding to an ambiguous keystroke sequence entered by a user. Each selection list entry appears directly on the face of a key. A user selects the desired entry from the selection list by pressing the associated key. Since sequential access to the selection list is no longer required, the select key of the preferred disambiguating system is assigned the function of a space key **1312**. In this embodiment it is useful, for ergonomic reasons, to associate a region **1314** of the touch-sensitive LCD with the first entry of the selection list. Designating region **1314** as a key corresponding to the first entry in the selection list allows the user to rapidly select the first entry using the thumb without moving his or her hands from the data keys.

A status window **1316** is also provided in the disambiguating system to display to the user the current state of the system. Indicator fields **1318**, **1320** in the status window show whether the next keystroke will be the first or second keystroke of a keystroke pair. In FIG. **12**, indicator field **1318** is "on" to indicate that the next keystroke is the first keystroke of a keystroke pair. An indicator field **1322** is also provided to indicate when the shift key has been depressed and that the next letter entered will be capitalized.

The embodiment of the reduced keyboard disambiguating system depicted in FIG. **12** operates in a similar manner to the preferred embodiment, but the touchscreen expands the flexibility of the system. Because each of the windows are touch sensitive, a user may select words or other symbols displayed in the windows by touching the LCD at the appropriate location. For example, a user can directly enter the editing mode by touching a word to edit in the text window **1306**. The faces of the keys in the keyboard **1304** may also be changed by the system to reflect different key interpretations. It therefore is not necessary to provide a key map in the text window to reflect different key meanings, since the faces of the keys can be directly changed to reflect the different meanings.

Variations can also be made in the construction and operation of each of the above embodiments. Those skilled in the art will appreciated that alternate arrangements exist for the keyboard **54** of the reduced keyboard disambiguating system. A 3×3 array of data keys **56** is preferred because the arrangement is easy to manipulate with the middle three fingers of a user's hand. All keystrokes are either on a center home row, or one row up or down. The system keys **58**, however, may be arrayed in different locations in order to speed text entry by the user. For example, as shown in a reduced keyboard disambiguating system **1400** in FIG. **13**, a select key **1402** may be located on the left-hand side of data keys **56**, and a delete key **1404** may be located on the right-hand side of the data keys. Locating the select and delete keys adjacent each side of the data keys simplifies operation of the keyboard since the user may use his or her thumb to operate the select key and his or her pinky to operate the delete key. The data keys **56** are operated by the middle three fingers of the user.

The 3×3 array of data keys **56** is also the preferred arrangement because a 3×3 array easily maps to other common user input devices other than keyboards. The user input device must be capable of moving to a number of states that correspond to the number of keys in the selected reduced keyboard. For example, a joystick can be substituted for the keyboard in the reduced keyboard disambiguating system. A keystroke is simulated by movement of the joystick. Moving the joystick in one of eight radial directions, each direction corresponding to a distinct state, and returning the joystick to a center default position corresponds to a keystroke on one of the outer circle of data keys. Pressing the joystick down while in the center default position corresponds to a keystroke on the center key of the data keys. A joystick button acts as a select key or other unambiguous system key. A user may therefore enter text using the joystick by moving the joystick to various states in order simulate the desired keystrokes. The disambiguating system would disambiguate the ambiguous keystrokes in the manner described above.

Other input devices can also be substituted for the keyboard in the reduced keyboard disambiguating system. As depicted in FIG. **14**, a control disk **1454** may be incorporated in a PDA **1452** or other portable electronic device. Control disk **1454** is a generally flat disk that may be tipped eight radial directions. Tipping the control disk in one of the eight radial directions generates a unique signal corresponding to the particular direction. The control disk is biased to return to a center default position after movement in a radial direction. Moving the control disk in one of the eight radial

directions and returning the control disk to the center default position is mapped to a keystroke on one of the outer circle of data keys. The control disk may also be pressed to generate a ninth unique signal. Pressing the control disk **1454** while in the center default position is mapped to a keystroke of the center key of the data keys. A dedicated select button **1456** and a dedicated delete button **1458** are locating on either side of the control disk. A user may therefore enter text using the control disk to simulate desired keystrokes. A disambiguating system **1450** connected to the control disk and incorporated in the PDA would disambiguate the ambiguous state sequence in the manner described above.

Another portable electronic device benefitting from the reduced size of the control disk is a watch. A watch could incorporate a small control disk on the face of the watch and the disambiguating system discussed above to allow a user to enter text messages into the watch. The messages may subsequently be downloaded or transferred to another system.

Several commercial user input devices are available that are particularly applicable for text entry using a mapping method like a joystick. For example, Interlink Electronics of Camarillo, California, manufactures a handheld pointing device called a ProPoint for use in audio-visual presentations. The pointing device is a generally flat disk that may be tipped radially to specify a direction, or pressed to act as a button. The pointing device also contains two dedicated buttons, one located above the control disk and the other located below the control disk. The device manufactured by Interlink may be readily adapted to operate with a disambiguating system by dividing the 360° movement of the control disk into eight segments, each of the segments mapped to a keystroke of one of the outer circle of data keys. Pressing the control disk corresponds to a keystroke of the center data key. The button located below the control disk corresponds to the select key, and the button located above the control disk corresponds to the delete key. When mapped in this manner, a user may rapidly input text using the technique described above.

A portable disambiguating system **1500** is shown in FIG. **15A**. The portable disambiguating system has a body **1502** that is generally shaped like a pen so that it may be easily grasped by a user. Protruding from an aperture **1504** at the tip of the body is an input shaft **1506**. One end of the input shaft **1506** is anchored inside the body by a ball and socket joint **1508**. The other end of the input shaft extends from the tip of the body, and terminates in a high-friction tip **1510**. The ball and socket joint within the body of the pen allows the input shaft to pivot about the joint.

The pivoting motion of the input shaft **1506** is constrained in two ways. First, a resilient bumper **1512** is positioned in the aperture at the tip of the disambiguating system body. The bumper surrounds the input shaft and contacts the input shaft near the tip **1510**. The bumper compresses as the input shaft pivots around the ball and socket joint to allow the input shaft to deflect from a default center position. When a pivoting force is removed from the input shaft, the bumper biases the input shaft to return to the default center position.

The pivoting motion of the input shaft is also constrained by an inflexible limiting disk **1514** which encircles the input shaft at a point between the bumper **1512** and the ball and socket joint **1508**. A cross-section of the limiting disk is shown in FIG. **15B**. The limiting disk **1514** is formed with eight cut-out segments **1516** that correspond to the shape of the input shaft. When the input shaft is moved outward from the default center position, the limiting disk limits the input shaft motion to one of eight radial directions corresponding to the cut-out segments.

Eight switches **1518** are radially positioned around the input shaft **1506** inside the body **1502** of the portable disambiguating system. The switches are oriented so that each switch corresponds to one of the eight radial directions of input shaft movement defined by the limiting disk, as shown in the cross-section of FIG. **15C**. Movement of the input shaft away from the default center position therefore brings the input shaft into contact with one of the eight switches. An additional switch is also located at the ball and socket joint **1508** of the input shaft. The additional switch is activated by a downward movement of the portable disambiguating system body, forcing the input shaft into the body of the disambiguating system.

The eight radial switches **1518** and the ninth switch at the ball and socket joint **1508** are connected to processing circuitry contained on a circuit card **1520** within the body of the portable disambiguating system. The processing circuitry includes a processor, associated memory, and a communications interface. Disambiguating software and vocabulary modules are stored within the processing circuitry to allow the disambiguating system to operate as a standalone device.

The eight radially mounted switches, and the ninth switch located at the ball and socket joint, are mapped using the technique described above to the nine data keys. One or more buttons **1522** are provided on the outer surface of the body **1502** to act as an unambiguous select and/or delete key. A user enters text with the portable disambiguating system **1500** by placing the tip **1510** of the input shaft on any available surface and moving the body of the pen around the input shaft to simulate keystrokes. The tip of the input shaft prevents the input shaft from sliding on the surface where it rests. Sufficient memory is provided on the circuit board **1520** so that text may be stored within the portable system and later downloaded to a computer. Alternatively, a direct connection may be made from the portable system to a computer or other electronic device having a display so that a user may view the text as it is being entered.

It will be appreciated that the portable system shown in FIG. **15A** may also include a display on the body **1502** of the device to allow the user to view the text as it is being entered. Preferably, the display would scroll text through the display window as the user enters text.

While the above discussion relates to the use of the reduced keyboard disambiguating system with Roman characters and English, it will be appreciated that the system is equally operable with foreign languages using different character sets. Foreign language versions would operate a similar manner, with the foreign character set grouped on the keys of the keyboard to optimize the keyboard and minimize the ambiguity in the particular language, or across multiple languages.

Those skilled in the art will also recognize that additional vocabulary modules can be enabled within the computer, for example vocabulary modules containing legal terms, medical terms, and foreign language terms. Via a system menu, the user can configure the system so that the additional vocabulary words can be caused to appear first or last in the list of possible words, with special coloration or highlighting. The disambiguation system can also be set so that the selection of such a word causes an audible warning beep.

Those skilled in the art will further recognize that the input device, in particular the keyboard, can be separated from the rest of the disambiguating system. In this embodiment, the portable device would comprise a key-

board and associated memory. Ambiguous keystrokes entered by a user are stored in the memory. The portable device is subsequently connected to the rest of the disambiguating system and the entered keystroke sequences disambiguated. This approach is particularly inexpensive since the portable input device is greatly simplified. Consequently, within the scope of the appended claims, it will be appreciated that the invention can be practiced otherwise than as specifically described herein.

The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. An input disambiguating system for disambiguating ambiguous input sequences entered by a user, the disambiguating system comprising:

(a) a user input device having a plurality of inputs, each of the plurality of inputs being associated with a plurality of characters, an input sequence being generated each time an input is selected by manipulating the user input device, with the generated sequence corresponding to the sequence of inputs that have been selected and having a textual interpretation that is ambiguous due to the plurality of characters associated with each input;

(b) a memory containing a plurality of objects, including textual objects that are completed words and textual objects that comprise a string of characters corresponding to the initial characters of an uncompleted word, each of the plurality of objects being associated with an input sequence and a frequency of use;

(c) an output device to provide system output to the user; and

(d) a processor coupled to the user input device, memory, and output device, the processor identifying from the plurality of objects contained in the memory at least one object associated with each generated input sequence, automatically selecting an object having a highest frequency of use from the identified objects associated with each generated input sequence, and generating an output signal causing the output device to provide the user the selected object as a textual interpretation of the entered input sequence.

2. The disambiguating system of claim 1, wherein the user input device further comprises an unambiguous selection input that generates a select signal when a user manipulates the user input device to activate said unambiguous selection input.

3. The disambiguating system of claim 1, wherein the user input device further comprises an unambiguous input and wherein the user may accept the object having the highest frequency of use as the textual interpretation of the entered input sequence by manipulating the user input device to activate an unambiguous input.

4. The disambiguating system of claim 2, wherein the processor further generates an output signal causing the output device to provide at least two of the identified objects associated with the same input sequence as possible textual interpretations of the entered input sequence.

5. The disambiguating system of claim 4, wherein the identified objects are provided to the user in order of decreasing frequency of use.

6. The disambiguating system of claim 3, wherein the user may select an alternate textual interpretation of the input sequence by repeated manipulations of the user input device to activate the unambiguous selection input, each manipulation of the user input device to activate the unambiguous input selecting a different object from the identified plurality of objects associated with the input sequence having a decreasing frequency of use.

7. The disambiguating system of claim 6, wherein at least two of the plurality of identified objects are displayed in a list, the list being truncated to omit objects that exceed a threshold number of objects to display, wherein said omitted objects are displayed in the list following one or more manipulations of the user input device to activate the unambiguous selection input.

8. An input disambiguating system for disambiguating ambiguous input sequences entered by a user, the disambiguating system comprising:

(a) a user input device having a plurality of inputs, each of the plurality of inputs being associated with a plurality of characters, an input sequence being generated each time an input is selected by manipulating the user input device and corresponding to the sequence of inputs that have been selected, the generated input sequence having a textual interpretation that is ambiguous due to the plurality of characters associated with each input;

(b) a memory containing a plurality of objects, each of the plurality of objects being associated with an input sequence and one of a plurality of object types, including a completed word type comprising textual objects that are completed words and a word stem type comprising textual objects that comprise a string of characters corresponding to the initial characters of an uncompleted word;

(c) a display; and

(d) a processor coupled to the user input device, the memory, and the display, wherein for each input in the input sequence entered by the user the processor identifies from the plurality of objects contained in the memory one or more objects associated with each generated input sequence and generates an output signal causing the display to display a selection list comprising the identified one or more objects, wherein if the processor identifies a plurality of objects comprising two or more object types, the selection list comprises identified objects of a plurality of object types.

9. The disambiguating system of claim 8, further comprising a speaker capable of emitting a plurality of tones connected to the processor, the processor generating a signal causing the speaker to emit a desired one of the plurality of tones based on the objects in the selection list.

10. The disambiguating system of claim 8, wherein each of the plurality of objects in the memory are further associated with a frequency of use.

11. The disambiguating system of claim 10, wherein a plurality of objects associated with a generated input sequence are displayed to the user in order of decreasing frequency of use.

12. The disambiguating system of claim 8, wherein each of the plurality of object types is further associated with a priority.

13. The disambiguating system of claim 12, wherein each of the plurality of objects in the selection list are ordered according to the priority associated with the object type associated with each object.

14. The disambiguating system of claim 8, wherein one of the plurality of object types is a character interpretation of an input sequence, the character interpretation interpreting at least one input in the input sequence as unambiguously specifying a character.

15. The disambiguating system of claim 8, wherein one of the plurality of object types is a proper noun interpretation of an input sequence, the proper noun interpretation asso-

ciating at least one character to at least one input in the input sequence and forming a completed proper noun.

16. The disambiguating system of claim 15, wherein the user input device further comprises an unambiguous shift input that generates a shift signal when the user input device is manipulated to activate the unambiguous shift input.

17. The disambiguating system of claim 16, wherein objects associated with the proper noun interpretation object type are displayed in the selection list before the objects associated with each of the other plurality of interpretations when the processor detects the shift signal before receiving an input sequence.

18. The disambiguating system of claim 8, wherein the user may select the plurality of object types that are displayed in the selection list.

19. The disambiguating system of claim 18, wherein the user may select a desired priority between the plurality of object types that are displayed in the selection list.

20. A text and pointing device to allow text entry into a computer coupled to a display, the text and pointing device comprising:

(a) a mouse coupled to said computer;

(b) a user input device incorporated in the mouse and having a plurality of inputs, each of the plurality of inputs being associated with a plurality of characters, an input sequence being generated each time an input is selected by manipulating the user input device and corresponding to the sequence of inputs that have been selected, the generated input sequence having a textual interpretation that is ambiguous due to the plurality of characters associated with each input; and

(c) an input disambiguating system coupled to the user input device for disambiguating ambiguous input sequences entered by the user, the disambiguating system comprising:

(1) a memory containing a plurality of objects, including textual objects that are completed words and textual objects that comprise a string of characters corresponding to the initial characters of an uncompleted word, each of the plurality of objects being associated with an input sequence; and

(2) a processor coupled to the user input device, memory, and said display, the processor identifying at least one of the plurality of objects in the memory associated with each generated input sequence, and generating an output signal causing the display to display at least one of the identified objects associated with each generated input sequence as a textual interpretation of the generated input sequence.

21. An input disambiguating system for disambiguating ambiguous input sequences entered by a user, the disambiguating system comprising:

(a) a user input device having a plurality of inputs, each of the plurality of inputs being associated with a plurality of characters, an input sequence being generated each time an input is selected by manipulating the user input device and corresponding to the sequence of inputs that have been selected, the generated input sequence having a textual interpretation that is ambiguous due to the plurality of characters associated with each input;

(b) a memory containing a plurality of objects, including word objects comprising textual objects that are completed words and word stem objects comprising textual objects that include a string of characters corresponding to the initial characters of an uncompleted word, each of the plurality of objects being associated with an input sequence;

(c) a display to depict system output to the user; and

(d) a processor coupled to the user input device, memory, and display, the processor identifying from the plurality of objects in the memory at least one object associated with each generated input sequence, and generating an output signal causing the display to display at least one of the identified objects associated with each generated input sequence as a textual interpretation of the generated input sequence.

22. The disambiguating system of claim 21, wherein each of the plurality of objects in the memory are further associated with a frequency of use.

23. The disambiguating system of claim 22, wherein the identified objects associated with the same input sequence are presented to the user in order of decreasing frequency of use.

24. The disambiguating system of claim 23, wherein the user input device further comprises an unambiguous selection input that generates a select signal when a user manipulates the user input device to activate said unambiguous selection input.

25. The disambiguating system of claim 23, wherein the user input device further comprises an unambiguous input and wherein an input sequence is terminated by the user manipulating the user input device to activate the unambiguous input.

26. The disambiguating system of claim 25, wherein the processor automatically selects an object having a highest frequency of use from the identified objects as the textual interpretation of the ambiguous input sequence upon detecting the manipulation of the user input device to activate an unambiguous input.

27. The disambiguating system of claim 24, wherein the user may select an object having a lower frequency of use as the textual interpretation of the ambiguous input sequence by manipulating the user input device to activate said unambiguous selection input a plurality of times, each manipulation of the user input device to activate said unambiguous input selecting an object having a next lower frequency of use from the identified objects.

28. The disambiguating system of claim 26, wherein the user may enter a second input sequence following the manipulation of the user input device to activate an unambiguous input.

29. The disambiguating system of claim 28, wherein a single space is appended to the selected textual interpretation of the received input sequence that is shown on the display following one or more manipulations of the user input device to activate said unambiguous selection input in an uninterrupted sequence.

30. The disambiguating system of claim 21, wherein each of the plurality of objects in memory belong to one of a plurality of object types, including a completed word type comprising textual objects that are completed words and a word stem type comprising textual objects that comprise a string of characters corresponding to the initial characters of an uncompleted word.

31. The disambiguating system of claim 30, wherein the identified objects associated with the received input sequence are displayed in a selection list on the display and wherein if a plurality of objects comprising two or more object types are identified, the selection list comprises identified objects of a plurality of object.

32. The disambiguating system of claim 31, further comprising a speaker capable of emitting a plurality of tones

connected to the processor, the processor generating a signal causing the speaker to emit a desired one of the plurality of tones based on the objects in the selection list and there associated types.

33. The disambiguating system of claim **31**, wherein each of the plurality of objects in the memory are further associated with a frequency of use.

34. The disambiguating system of claim **33**, wherein the objects in the selection list are presented to the user in order of decreasing frequency of use.

35. The disambiguating system of claim **31**, wherein each of the plurality of object types is further associated with a priority.

36. The disambiguating system of claim **35**, wherein each of the plurality of objects in the selection list are ordered according to the priority associated with the object type associated with each object.

37. The disambiguating system of claim **31**, wherein one of the plurality of object types is an unambiguous character interpretation of an input sequence, the unambiguous character interpretation interpreting the input sequence as unambiguously specifying a sequence of one or more characters.

38. The disambiguating system of claim **31**, wherein one of the plurality of object types is a proper noun interpretation of an input sequence, the proper noun interpretation associating at least one character to at least one input in the input sequence and forming a completed proper noun.

39. The disambiguating system of claim **38**, wherein the user input device further comprises an unambiguous shift input that generates a shift signal when the user input device is manipulated to activate the unambiguous shift input.

40. The disambiguating system of claim **39**, wherein objects associated with the proper noun interpretation object type are displayed in the selection list before the objects associated with each of the other plurality of interpretations when the processor detects the shift signal before receiving an input sequence.

41. The disambiguating system of claim **31**, wherein the user may identify the plurality of object types that are displayed in the selection list.

42. The disambiguating system of claim **41**, wherein the user may select a desired priority between the identified plurality of object types that are displayed in the selection list.

43. The disambiguating system of claim **21**, wherein the user input device is movable to a plurality of radial directions, each of the plurality of radial directions associated with one of the plurality of inputs.

44. The disambiguating system of claim **43**, wherein the user input device is movable to eight radial directions.

45. The disambiguating system of claim **44**, wherein the user input device is a control disk.

46. The disambiguating system of claim **21**, wherein the plurality of objects are stored in the memory in a tree comprised of a plurality of nodes, each node being associated with an input sequence and with a plurality of objects wherein word objects and word stem objects that are associated with the same input sequence are stored at the same node.

47. The disambiguating system of claim **46**, wherein the plurality of nodes are connected by a plurality of paths, each of the plurality of paths linking a parent node associated with a base input sequence with a child node associated with the base input sequence of the parent node and an additional input.

48. The disambiguating system of claim **47**, wherein the objects associated with a child node are based on the objects associated with the corresponding parent node to which the child node is linked.

49. The disambiguating system of claim **48**, wherein the objects associated with a child node are constructed using a code to modify objects associated with the corresponding parent node.

* * * * *

EXHIBIT C

US006185295B1

(12) **United States Patent**
Frederiksen et al.

(10) **Patent No.: US 6,185,295 B1**
(45) **Date of Patent: Feb. 6, 2001**

(54) **PHONE NUMBER DATABASE FOR A PHONE**

(75) Inventors: **Steen Lillethorup Frederiksen**, Allerød; **Peter Ib**, Herlev, both of (DK); **Erik Paul Anderson**, Turku (FI)

(73) Assignee: **Nokia Mobile Phones Limited**, Espoo (FI)

(*) Notice: Under 35 U.S.C. 154(b), the term of this patent shall be extended for 0 days.

(21) Appl. No.: **09/025,348**

(22) Filed: **Feb. 18, 1998**

(30) **Foreign Application Priority Data**

Feb. 21, 1997 (GB) 9703643

(51) **Int. Cl.**[7] **H04M 11/00**; H04M 1/274
(52) **U.S. Cl.** **379/355**; 379/216; 455/564
(58) **Field of Search** 379/354, 355, 379/356, 216; 455/564, 565, 558

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,134,717	*	7/1992	Rasmussen	455/89
5,267,308	*	11/1993	Jokinen et al.	379/354
5,363,437		11/1994	Shen et al.	379/355
5,491,745	*	2/1996	Roeder	379/355
5,509,067		4/1996	Murata	379/355
5,535,258	*	7/1996	Joglekar et al.	379/58
5,675,630	*	10/1997	Beatty	379/59
5,687,216	*	11/1997	Svensson	379/58
5,692,032		11/1997	Seppanen et al.	379/59
5,812,946	*	9/1998	Nakabayashi et al.	455/426
5,963,875	*	10/1999	Go	455/564
5,963,876	*	10/1999	Manssen et al.	455/564
5,991,396	*	11/1999	Salm et al.	379/355

FOREIGN PATENT DOCUMENTS

0 709 996 A2 5/1996 (EP) .

* cited by examiner

Primary Examiner—David R. Hudspeth
Assistant Examiner—Abul K. Azad
(74) *Attorney, Agent, or Firm*—Perman & Green, LLP

(57) **ABSTRACT**

A phone number database has a first group of memory locations with associated speed dialing facility and a second group of memory locations without speed dialing facility. A phone number may be provided with speed dialing facility when this phone number is already stored without speed dialing facility in the second group by indicating the phone number whose speed dialing status is to be changed, indicating a memory location with a phone number with the desired speed dialing status, and interchanging the speed dialing status of the two phone numbers.

12 Claims, 3 Drawing Sheets

Memory location no.	Name	Phone no.
1	voice mail box	+45 50 50 50 50
2	Gill	+46 59 3111 45
3	Jane	+49 89 600 0662
4	Carl	+44 1233 4561

• • •

65	Freddy	+43 1 789 1233
66	Bob	+32 2 323 1233
67	John	+39 35 200 222
68	Ben	+1 123 123 3123
69	Jackie	+358 0 444 444

• • •

Fig.1.

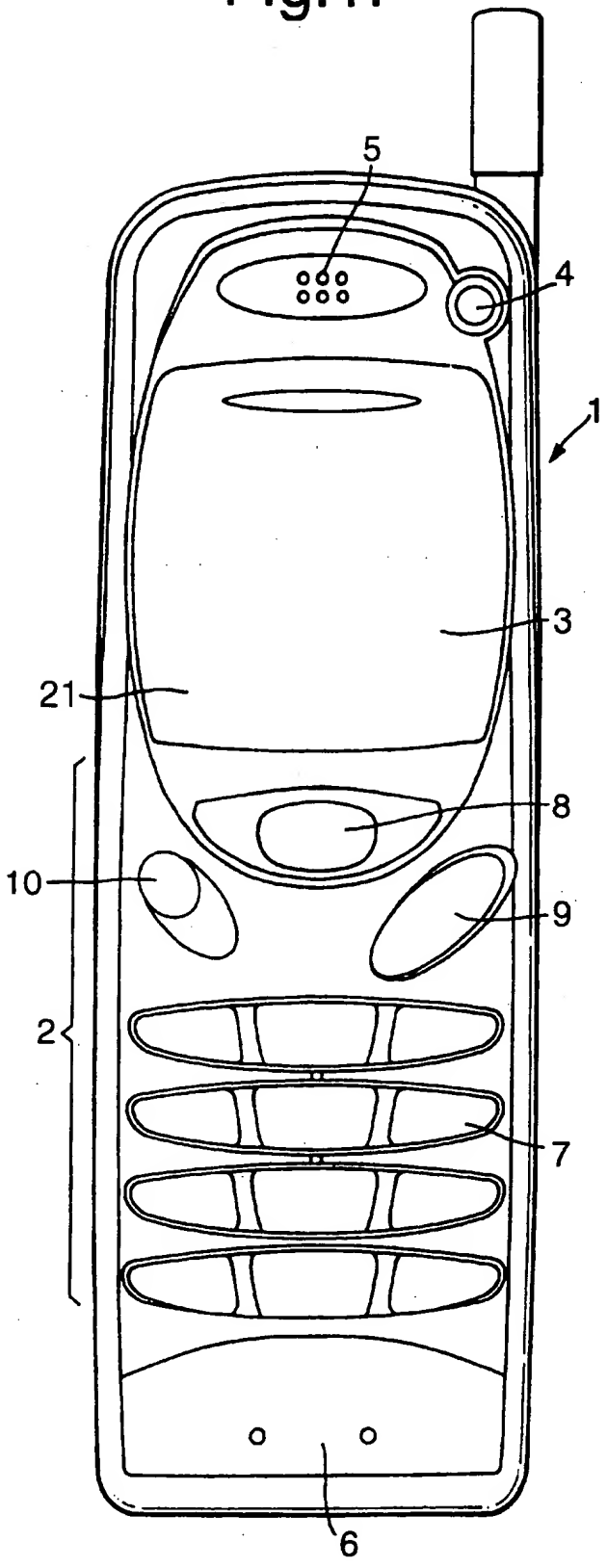


Fig.2.

Memory location no.	Name	Phone no.
1	voice mail box	+45 50 50 50 50
2	Gill	+46 59 3111 45
3	Jane	+49 89 600 0662
4	Carl	+44 1233 4561
• • •		
65	Freddy	+43 1 789 1233
66	Bob	+32 2 323 1233
67	John	+39 35 200 222
68	Ben	+1 123 123 3123
69	Jackie	+358 0 444 444
• • •		

Fig.3.

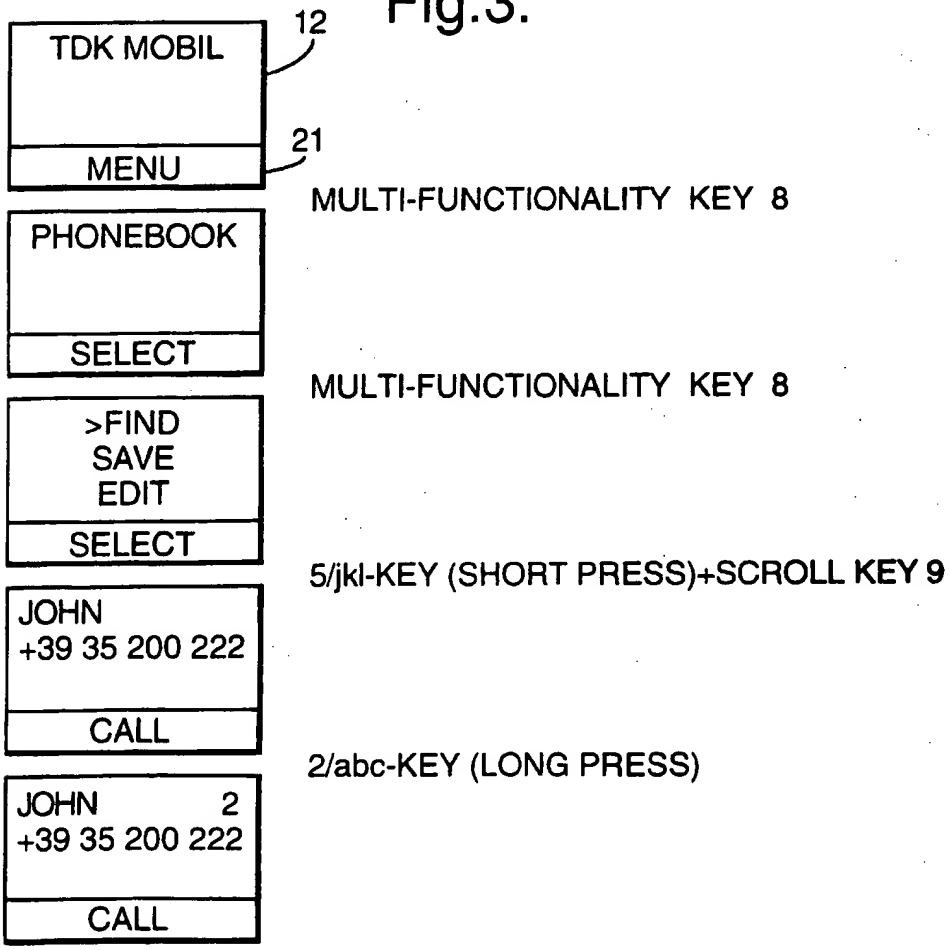
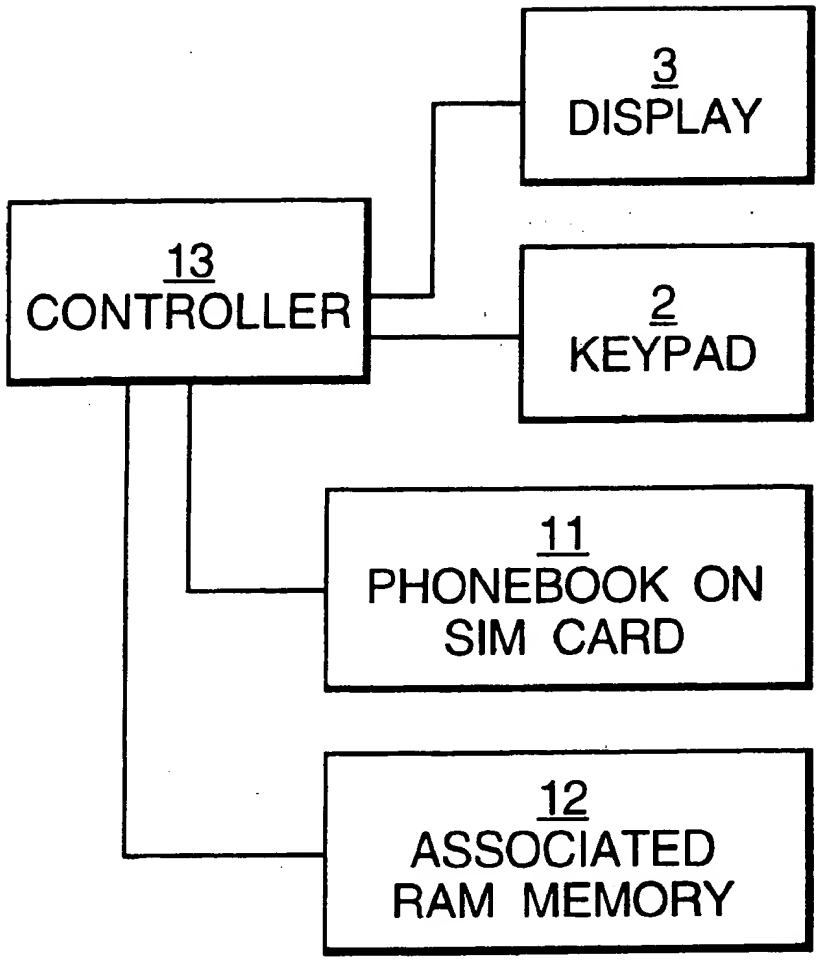


Fig.4.


3
DISPLAY
13
CONTROLLER
2
KEYPAD
11
PHONEBOOK ON
SIM CARD
12
ASSOCIATED
RAM MEMORY

PHONE NUMBER DATABASE FOR A PHONE

BACKGROUND OF THE INVENTION

The invention relates to a method of transferring a phone number already stored in a phone number database of a phone from a memory location without speed dialing facility to a memory location with speed dialing facility, without this causing loss of information already saved. The invention moreover relates to a phone for performing the method.

Hand-portable phones give the user the possibility of storing a plurality of phone numbers electronically. The actual storage may take place in a RAM in the phone, but will typically take place in a memory on a SIM card, and the resulting database will follow the SIM card and thus the phone subscription instead of the phone. The user may typically save 50–100 phone numbers having up to 30 digits and an associated name of about letters. These data may vary from SIM card supplier to SIM card supplier. The phone numbers will typically be stored in serially numbered memory locations, e.g. 1–100. A group of phone numbers—typically the numbers stored in the memory locations 2–9—will be accessible as short dialing numbers, which means that in the idle mode of the phone the user can make a call to a phone number stored in e.g. location #3 by depressing the "3" key for e.g. 0.5–1.0 second.

This feature, convenient and very useful to the user, is available only to a small group of numbers in the total number database. It may frequently be expedient to add new numbers to the group of speed dialing numbers, since the phone numbers most frequently called frequently change. It will be possible to overwrite an existing memory location in the speed dialing number group by adding a new number, but this will involve loss of data, which will thus have to be entered again. Further, it is not desired to extend the group of speed dialing numbers, since it will be too difficult to remember the short dialing code for speed dialing numbers used less frequently.

SUMMARY OF THE INVENTION

The invention comprises a method of providing a phone number with speed dialing facility when said phone number is already stored without speed dialing facility in a phone number database, which has a first group of memory locations with associated speed dialing facility and a second group of memory locations without speed dialing facility, said method comprising indicating the phone number whose speed dialing status is to be changed, indicating a memory location with a phone number having the desired speed dialing status and interchanging the speed dialing status of the two phone numbers.

A phone number may hereby be given a desired speed dialing status by a simple operation, the only expense to the database being that a phone number already having this status merely loses it. No information is lost or has to be entered again.

The invention provides a method enabling the user with few instructions to transfer numbers already stored to the group of speed dialing numbers without losing information already saved.

In a preferred embodiment, the contents of one memory location are saved temporarily, and then the contents of the other memory location are transferred to the first location. The contents are then transferred from the intermediate memory to the other location. The interesting point of swapping the contents of the two memory locations is that the information is maintained, it being just the phone numbers which change status because of their shift in memory location.

When the phone book is entered according to the preferred embodiment, one or more items will be displayed on the display. The phone will have a scroll key by means of which the user may scroll through the list of items. The selected item will be indicated, optionally highlighted, on the display. When the user has found the item whose status he wishes to change, he is to select a destination address. As the group of memory locations with associated speed dialing facility is stored on the locations 2–9, the destination memory location is expediently selected by long depression of a one of the keys 2–9. If, e.g. the memory locations 2–19 had the speed dialing facility, the location 18 might be selected as the destination memory location by ordinary entering of "1", while "8" was depressed for an extended period of time (long press).

The invention moreover relates to a telephone, radio telephone or handset having a phone number database having a first group of memory locations with associated speed dialing facility and a second group of memory locations without speed dialing facility. Such a phone number database may be called an electronic phone book, and it is the user of the phone who handles the contents of the database. The phone has a display on which at least a part of the contents of the of the phone number database may be displayed, a cursor for visual identification of an item on the display and a positioning device for moving the cursor between items on the display. Further, the phone has another identification device for selecting a memory location among the first group of memory locations with associated speed dialing facility, and a device for interchanging the speed dialing status of the two identified phone numbers. The preferred embodiment makes it possible, in a particularly simple procedure, to update the contents of the group of memory locations with associated speed dialing facility by transferring information already stored to one of these memory locations, without losing the information originally contained in the memory location concerned. Instead, this information is saved in the memory location whose contents have just been transferred to the group of memory locations with associated speed dialing facility.

Swapping of the contents of the two memory locations takes place in practice in that an intermediate memory receives the contents of the memory location to be transferred to the group of memory locations with associated speed dialing facility. This location is identified with a cursor in the display. The controller of the phone then transfers the contents of the memory location which has till now belonged to the group of memory locations with associated speed dialing facility to the memory location whose contents have just been transferred to the intermediate memory. When this has been done, the controller transfers the contents of the intermediate memory to the memory location which was selected as the reception location for the phone number which is to be provided with a speed dialing facility.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will be explained more fully below in connection with a preferred embodiment and with reference to the drawing, in which:

FIG. 1 shows a preferred embodiment of a portable phone having a user interface according to the invention;

FIG. 2 schematically shows an example of the contents of a phone book stored on the SIM card of the phone;

FIG. 3 shows a sequence of display images in connection with swapping of memory locations in the phone book; and

FIG. 4 illustrates the cooperation between the controller, the SIM card memory and an intermediate memory according to a preferred embodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION

FIG. 1 shows a preferred embodiment of a radiophone according to the invention, and it will be seen that the phone, which is generally designated by 1, comprises a user interface having a keypad 2, a display 3, an on/off button 114, an earpiece 5, and a microphone 6. The keypad 2 has a first group 7 of keys in the form of alphanumeric keys, by means of which the user can enter a phone number, write a text message (SMS), write a name (associated with the phone number), etc. The user uses the first group of keys primarily for entering data in the phone (entry events).

The keypad 2 additionally comprises a second group of keys which, in the preferred embodiment, comprises precisely one multi-functionality key 8 or soft key whose function depends on the present state of the phone. The default function or the present function of the multi-functionality key 8 is displayed in a predetermined area 21 of the display 3. In the preferred embodiment, the second group of keys additionally comprises a scroll key 9 by means of which the user can jump selectively from one item to the preceding or the succeeding item in the menu loop of the phone, while he gets access to a submenu loop under the item concerned in the main menu loop by activation of the operation key. The clear key 10 may be used e.g. for erasing the digit or letter entered last by brief depression, while depression of a longer duration will erase the entire number of word. Like the multi-functionality key 8, the scroll key 9 and the clear key 10 may advantageously be redefined in some states, which appears from the following.

A preferred embodiment of the invention will be described below with reference to FIGS. 2–4. In the preferred embodiment, a controller 13 may be formed by the physical layer processor of the phone which, as will be known to a skilled person, monitors i.a. the cellular network, handles the call establishment and otherwise controls the user interface of the phone. The controller 13 is thus connected to both the keypad 2 and the display 3.

Further, the controller 13 controls the memories of the phone, including the storage of phone numbers in the phone book memory, which is an external memory 11 on the SIM card of the phone in the preferred embodiment. This memory is designed so as to be addressed by a memory location number, and, for each phone number, is capable of accommodating a plurality of signs, e.g. 14, for the identification of the phone number proprietor, a plurality of signs, e.g. 30, for the phone number itself, said memory 11 typically having 50–100 memory locations. However, these amounts vary considerably from card supplier to card supplier.

Several internal memories 12 are provided in connection with the controller 13 for the temporary storage of information, which is entered with the keypad 2, is shown on the display 3 or is received via the cellular network.

The structure of the memory 11 on the SIM card is illustrated in FIG. 2. The card supplier usually stores the number of a mail box service on memory location 1, for which reason the memory locations 2–9 may be provided with speed dialing facility, as the user will be able to make a call in the idle mode of the phone to the number stored in memory location #4 merely by e.g. depressing the figure "4" for 0.8 second.

FIG. 2 schematically shows a section of a phone book memory 11 in which the memory locations #1–4 and #65–69 are shown, while the rest is left out for clarity. Names as well as numbers are purely fictitious and just serve purposes of illustration. Usually, the user of the phone builds up the phone book database through occasional entering of phone numbers. Since, at the same time, the numbers most frequently called from a phone change dynamically, there is a need for a simple and effective way of changing the status of already stored numbers with respect to speed dialing. The preferred embodiment of the invention will be explained below on the basis that the user wants "John"'s phone number, stored in memory location #67, to be provided with speed dialing facility. The user wants this speed dialing facility to be obtained from "Gill"'s memory location (memory location #2).

As shown, this procedure proceeds with the five display images shown in FIG. 3 in the preferred embodiment. In the idle mode of the phone, the display image 3 contains information on the network operator concerned, which is TDK Mobil here. Further, a predetermined part 21 of the display contains information on the present functionality of the multi-functionality key 8, which is here access to the main menu of the phone. The phone book is the first item in the main menu loop, and the user, with a single activation of the multi-functionality key 8, therefore arrives at the phone book with the display image shown as the second image in the sequence. It is indicated in the display that the phone book may be selected by depressing the multi-functionality key 8. If the user does not wish to select the phone book, he may move further on in the main menu loop by means of the scroll key 9. By depressing the multi-functionality key 8 once more, the phone enters a phone book mode, it being possible for the user to select a phone number handling option, such as "Find", "Save" and "Edit" shown in the third display image. The user may here run through the entire group of options by means of the scroll key 9 and select an option by means of the multi-functionality key 8 when the option is indicated by the ">" sign. The phone arranges the stored names alphabetically, and the user may therefore scroll through the alphabetic name list by means of the scroll key 9. This sorting does not take the memory location number into consideration, and, therefore, the user is only informed of this for numbers which have speed dialing status. The alphabetic name list is endless, i.e. turns around after the last item on the list. Only the location addresses of phone numbers having the speed dialing facility are shown on the display.

The user may get into the alphabetic name list in several ways. Firstly, by selecting "Find" and entering the name "John", he may arrive directly at "John"'s phone number. Alternatively, when the phone book has been selected and the phone number handling options have been presented, the user may depress one of the alphanumeric keys, causing the first name corresponding to the alphanumeric value of the key to be displayed on the display. If there are no other names beginning with "J" in the phone book, "Jane" will be the first name which is shown after the activation of the "5/jkl" key, and depression of the scroll key 9 in the scroll-down direction will cause "John" to be shown immediately afterwards, as is the case here, or after a number of depressions if there are several names between the two in the phone book.

The indication of the phone number whose speed dialing status is to be changed takes place in the preferred embodi-

ment in that the number is shown solely on the display. The display thus serves as a cursor in the preferred embodiment. Alternatively, the number might be indicated e.g. by highlighting it among a group of numbers from the phone book shown on the display.

In connection with the indication of "John", the default function of the multi-functionality key **8** will be to make a call to "John". When, instead, the "2/abc" key is depressed for more than e.g. 0.8 sec. (long press), the memory location #2 will be selected as destination memory location (and thereby indicated by a memory location with a phone number having the desired speed dialing status) for the contents of the memory location shown (here memory location #67). In the preferred embodiment, the actual interchange of the speed dialing status of the two phone numbers takes place by swapping the numbers between the two memory locations.

When "John"'s phone number is shown on the display, a copy of memory location #67 may already have been saved in the RAM **12** for intermediate storage of the contents of one of the two indicated memory locations. The controller **13** will hereby be able to transfer the contents of the destination memory location (here memory location #2) to the memory location having the phone number which is to be given speed dialing facility. When this has been done, the contents of the memory location #67, now just saved in the RAM **12**, may be transferred finally to the destination memory location, whereby the swapping of the two phone numbers is completed. As the location address #2 for "John"' phone number has speed dialing facility, this is indicated on the display in the last image in the sequence by the display of a "2" figure, which also confirms that the swapping has been completed successfully.

It should be noted that the period of time separating long press and short press may vary within wide limits, but 0.8 sec. +/−0.4 sec. will usually be preferred. It should moreover be noted that long press of a number key in the phone book mode will cause swapping of the two numbers, while, in idle mode, it will cause calling the number stored on the speed dialing address concerned. Finally, depression of the number key in the phone book mode for a shorter duration of time will cause displaying of another number instead of swapping of two numbers. Thus, it is clear that the order is carried out only when the key is released.

The swapping may also be performed even if the two indicated memory locations do not contain phone numbers.

Alternatively, the contents of the destination memory location may be saved temporarily in the RAM **12**, following which the controller **13** transfers the contents of the memory location having the phone number to be given speed dialing facility to the destination memory location (here memory location #2). When this has been done, the contents of the memory location #67, now just saved in the RAM **12**, may be transferred finally to the destination memory location, whereby the swapping of the two phone numbers is completed.

The present invention includes any novel feature or combination of features disclosed herein either explicitly or any generalization thereof irrespective of whether or not it relates to the claimed invention or mitigates any or all of the problems addressed.

In view of the foregoing description it will be evident to a person skilled in the art that various modifications may be made within the scope of the invention.

What is claimed is:

1. A method of handling speed dialing status of phone numbers stored in a phone number database, having a first group of memory locations being associated with speed dialing facilities and a second group of memory locations without speed dialing facilities, the method comprising the steps of:

selecting a first phone number whose speed dialing status is to be changed wherein the first number is stored in a first memory location in one of the first and second groups of memory locations;

designating a second phone number, wherein the second phone number is stored in a second memory location having the desired speed dialing status in another one of the first and second groups of memory locations; and

swapping a content of the first memory location with a content of the second memory location in order to interchange the speed dialing status of the two phone numbers, the swapping of the content of the two memory locations comprising the steps of:

temporarily storing the contents of one of the two selected memory locations in an intermediate memory;

transferring the contents of the other of the two memory locations to the memory location whose contents are temporarily stored in the intermediate memory; and

transferring the contents of the intermediate memory to the memory location whose contents have been transferred to the other memory location.

2. A method according to claim **1**, wherein the step of selecting a first phone number whose speed dialing status is to be changed comprises:

finding and displaying the phone number whose speed dialing status is to be changed in a phone number list from the phone number database; and

pressing a number key corresponding to the designated member location address.

3. A method according to claim **2**, wherein phone numbers in memory locations having a one-digit address may be called by pressing a number key corresponding to the one digit address for a first predetermined period of time when the phone is in an idle mode, and wherein a designation of the memory location with the phone number having the desired speed dialing status comprises depressing a number key corresponding to the address for a second predetermined period of time.

4. A method according to claim **3**, wherein phone numbers in memory locations having one-digit addresses may be called by continuous depression of a number key corresponding to the address for a first predetermined period of time when the phone is in idle mode, and wherein indication of the memory location with the phone number having the desired speed dialing status comprises depressing a number key corresponding to the address for a second predetermined period of time.

5. A method according to claim **4**, wherein the first predetermined period of time corresponds to the second predetermined period of time.

6. A phone with a phone number database having a first group of memory locations being associated with speed dialing facilities and a second group of memory locations without speed dialing facilities, and moreover comprising;

an input means for inputting instructions into the phone;

a display for displaying information;

a controller connected to the input means and controlling the display of information;

wherein the input means comprises:

means for presenting the content of a first memory location in the display, wherein the phone number

stored in the first memory location is desired to become associated with the speed dialing facility;

means for designating a memory location having the desired speed dialing facility; and

means for swapping the content of the two memory locations in order to interchange the speed dialing status of the two phone numbers; and

wherein the phone further includes:

an intermediate memory for storing the contents of the first memory location whose content is presenting in the display;

means for transferring the contents of the other of the two indicated memory locations to the memory location whose contents are temporarily stored in the intermediate memory; and

means for transferring the contents of the intermediate memory to the memory location whose contents have been transferred to the other memory location.

7. A phone according to claim **6**, wherein the visual identification of an item on the display comprises:

means for displaying at least a part of the items of the phone number database on the display; and

a positioning device in the form of a scroll key for stepwise movement of the cursor between items on the display for indicating the phone number whose speed dialing status is to be changed.

8. A phone according to claim **6**, wherein the identification device for selecting a memory location among the first group of memory locations with associated speed dialing facility comprises a numeric keypad, and wherein the selection takes place by activating the keys of the keypad corresponding to the address of the memory location concerned.

9. A phone according to claim **8**, wherein the first group of memory locations with associated speed dialing facility has one-digit addresses in the phone number database and may be called by continuous depression of a number key corresponding to the address for a first predetermined period of time when the phone is in idle mode, and wherein the indication of the memory location with the phone number with the desired speed dialing status comprises depression of a number key corresponding to the address for a second predetermined period of time.

10. A phone according to claim **9**, wherein the duration of the first predetermined period of time corresponds to the duration of the second predetermined period of time.

11. A phone according to claim **6**, wherein the means for designating a memory location among the first group of memory locations with associated speed dialing facility comprises a numeric keypad and wherein the selection takes place by pressing the numeric key corresponding to the address of the designated memory location for a period with a predetermined duration.

12. A phone according to claim **6**, wherein the first group and the second group are present in a separate memory of a replaceable subscriber identity module (SIM) card.

* * * * *

EXHIBIT D

US005797098A

United States Patent [19]

Schroeder et al.

[11] Patent Number: 5,797,098

[45] Date of Patent: Aug. 18, 1998

[54] **USER INTERFACE FOR CELLULAR TELEPHONE**

[75] Inventors: **Martin K. Schroeder**, San Diego, Calif.; **Duane Sharman**, Calgary, Canada

[73] Assignee: **Pacific Communication Sciences, Inc.**, San Diego, Calif.

[21] Appl. No.: **504,121**

[22] Filed: **Jul. 19, 1995**

[51] **Int. Cl.**[6] **H04M 11/00**; H04Q 7/00

[52] **U.S. Cl.** **455/464**; 379/355

[58] **Field of Search** ..... 379/59, 58, 58 F, 379/63, 63 F, 355; 455/33.1, 564, 566; 395/794

[56] **References Cited**

U.S. PATENT DOCUMENTS

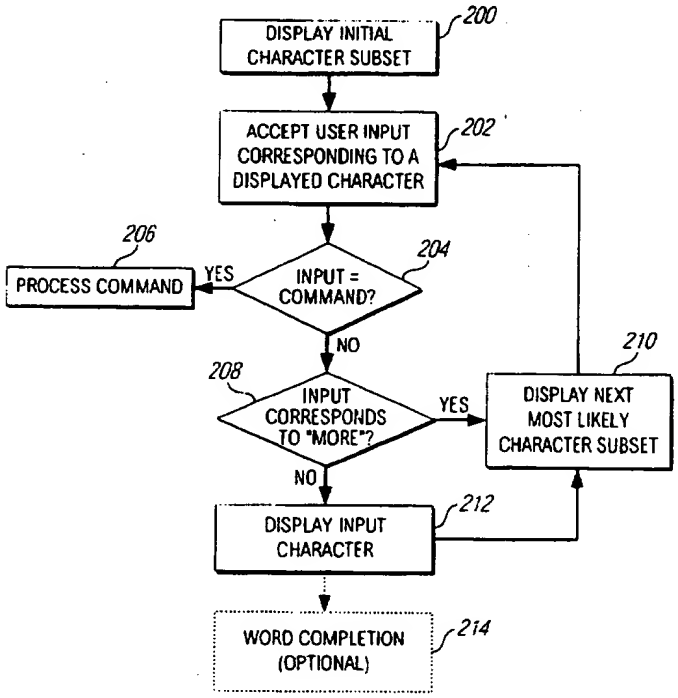
4,158,236	6/1979	Levy	395/794
4,661,916	4/1987	Baker et al.	364/513.5
4,731,811	3/1988	Dubus	379/58
4,747,153	5/1988	Yoshimura et al.	395/794
4,837,797	6/1989	Freeny et al.	379/96
5,097,425	3/1992	Baker et al.	395/2
5,128,672	7/1992	Kaehler	341/23
5,153,904	10/1992	Coombes et al.	379/58
5,297,041	3/1994	Kushler et al.	364/419.15
5,301,227	4/1994	Kamei et al.	379/88
5,329,578	7/1994	Brennan et al.	379/67
5,367,558	11/1994	Gillig et al.	379/59
5,384,825	1/1995	Dillard et al.	455/564
5,398,115	3/1995	Lin	358/407
5,402,481	3/1995	Waldman	379/355
5,459,774	10/1995	Breeden	455/566
5,475,743	12/1995	Nixon et al.	379/113
5,491,745	2/1996	Roeder	455/564
5,500,859	3/1996	Sharman et al.	370/112
5,511,111	4/1996	Serbetcioglu et al.	379/67
5,535,260	7/1996	Zicker et al.	379/63

Primary Examiner—David R. Hudspeth
Assistant Examiner—Scott Richardson
Attorney, Agent, or Firm—John F. Land; Robert P. Sabath; Steven A. Shaw

[57] **ABSTRACT**

An improved user interface for a cellular telephone system subscriber unit, including the following functions: (1) a predictive keyboard input method for speeding up input on a telephone with a space limited keyboard; (2) a word completion method for speeding up input; (3) a distinctive signaling method useful in a dual-mode or tri-mode cellular phone system that incorporates both voice call functionality and data messaging functionality; (4) a secret message method that permits secret messages to be received by an authorized user of a cellular telephone that includes a data messaging capability; (5) a message screening method that permits a user to set a message screening mode in a cellular telephone; (6) an improved "scratchpad" method which permits a user to enter a telephone number into a storage register of a cellular telephone while in the middle of a voice call, visually verify the entry, and then save the number to a rapid redial location for later use; (7) a global search method for searching text strings in all of the different memory sections of a cellular telephone having an address book, a hierarchical menu structure, and stored data messages.

11 Claims, 14 Drawing Sheets



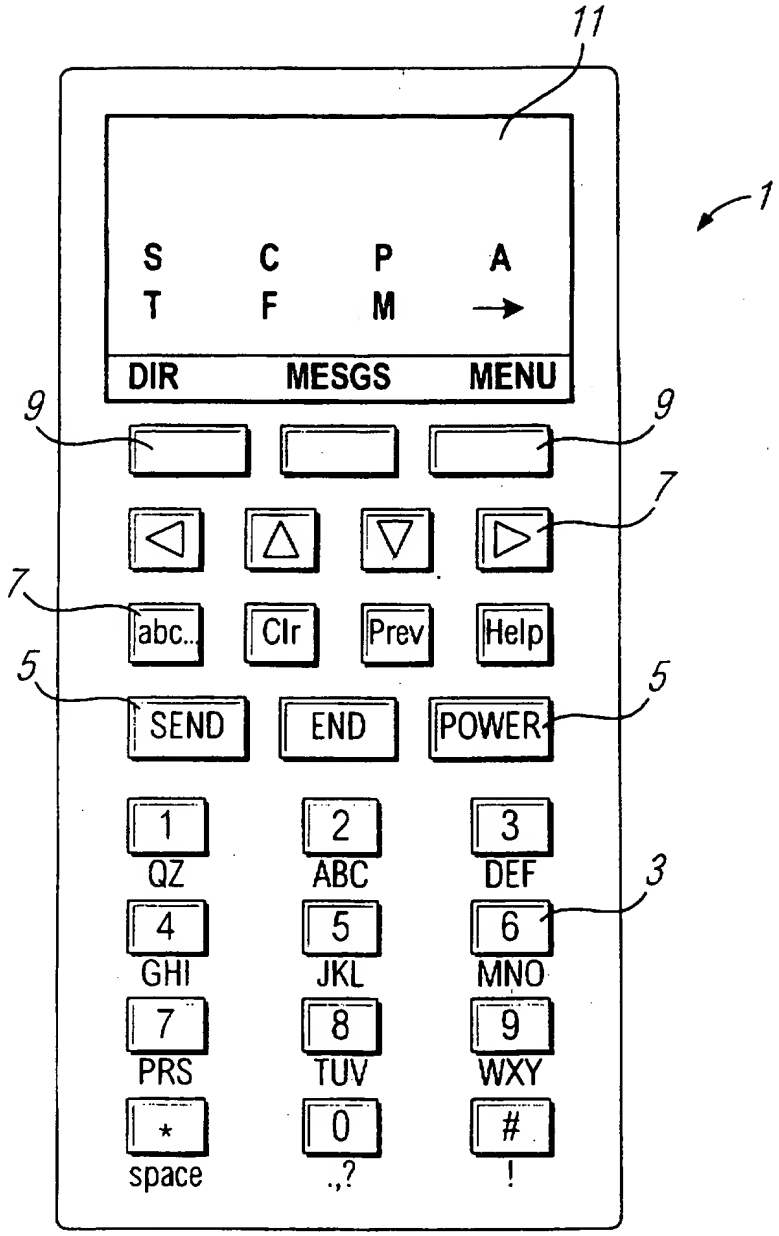

11
1
S C P A
T F M
DIR MESGS MENU
9
9
7
7
abc..
Clr
Prev
Help
5
SEND
END
POWER
5
1 QZ
2 ABC
3 DEF
3
4 GHI
5 JKL
6 MNO
7 PRS
8 TUV
9 WXY
* space
0 .,?
!

FIG. 1A

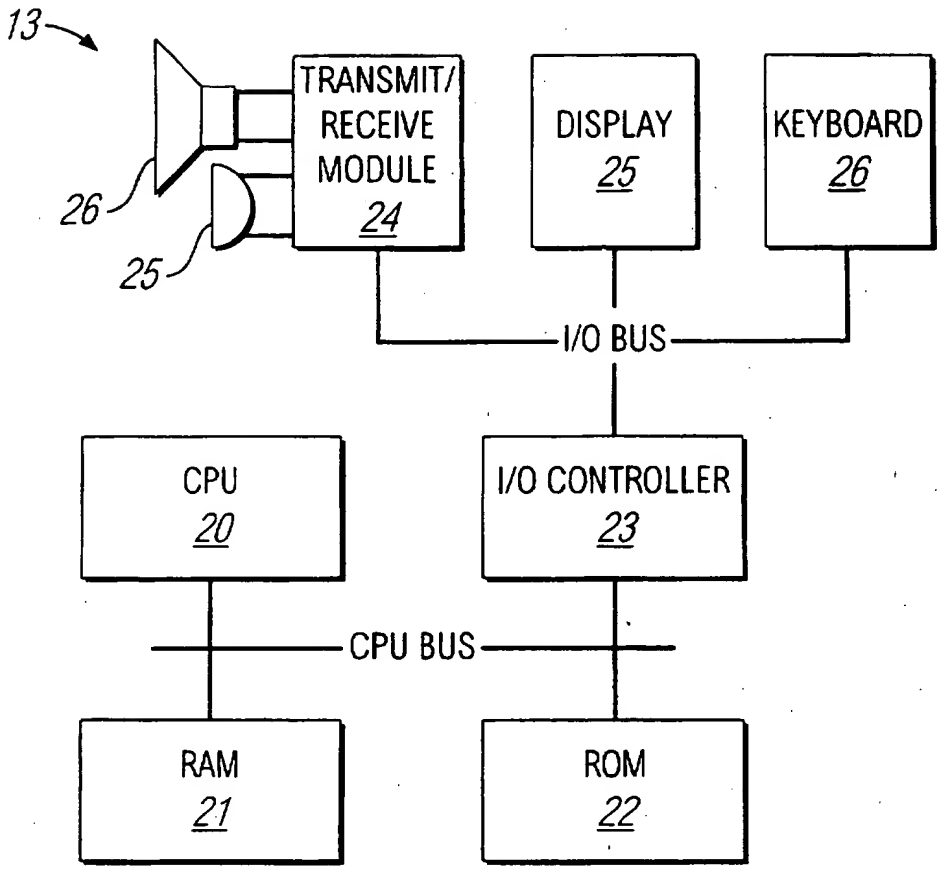

13
TRANSMIT/
RECEIVE
MODULE
24
DISPLAY
25
KEYBOARD
26
26
25
I/O BUS
CPU
20
I/O CONTROLLER
23
CPU BUS
RAM
21
ROM
22

FIG. 1B

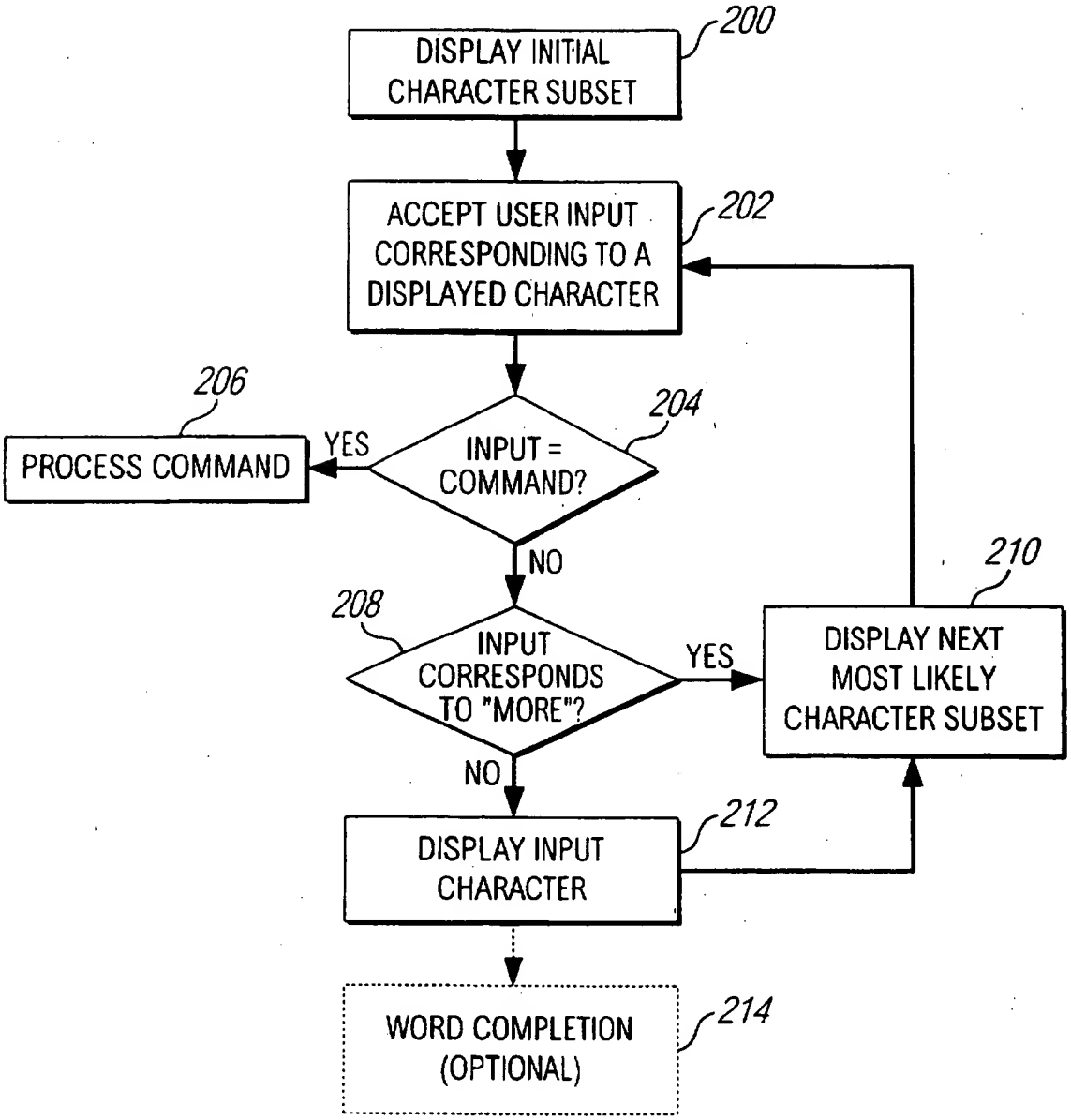

DISPLAY INITIAL CHARACTER SUBSET
200
ACCEPT USER INPUT CORRESPONDING TO A DISPLAYED CHARACTER
202
INPUT = COMMAND?
204
YES
PROCESS COMMAND
206
NO
INPUT CORRESPONDS TO "MORE"?
208
YES
DISPLAY NEXT MOST LIKELY CHARACTER SUBSET
210
NO
DISPLAY INPUT CHARACTER
212
WORD COMPLETION (OPTIONAL)
214

FIG. 2

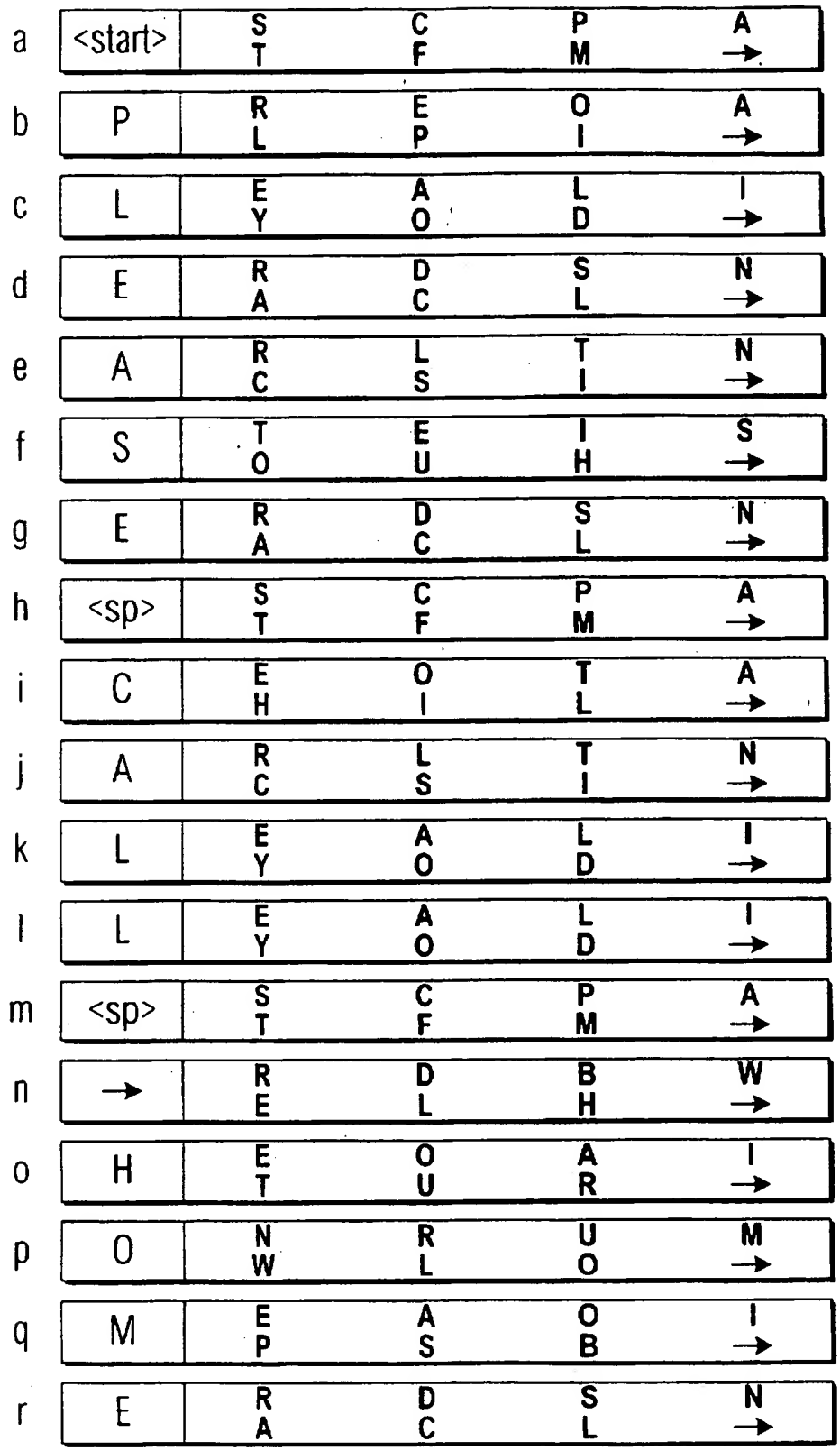

a
<start>
S T
C F
P M
A →
b
P
R L
E P
O I
A →
c
L
E Y
A O
L D
I →
d
E
R A
D C
S L
N →
e
A
R C
L S
T I
N →
f
S
T O
E U
I H
S →
g
E
R A
D C
S L
N →
h
<sp>
S T
C F
P M
A →
i
C
E H
O I
T L
A →
j
A
R C
L S
T I
N →
k
L
E Y
A O
L D
I →
l
L
E Y
A O
L D
I →
m
<sp>
S T
C F
P M
A →
n
→
R E
D L
B H
W →
o
H
E T
O U
A R
I →
p
O
N W
R L
U O
M →
q
M
E P
A S
O B
I →
r
E
R A
D C
S L
N →

FIG. 3

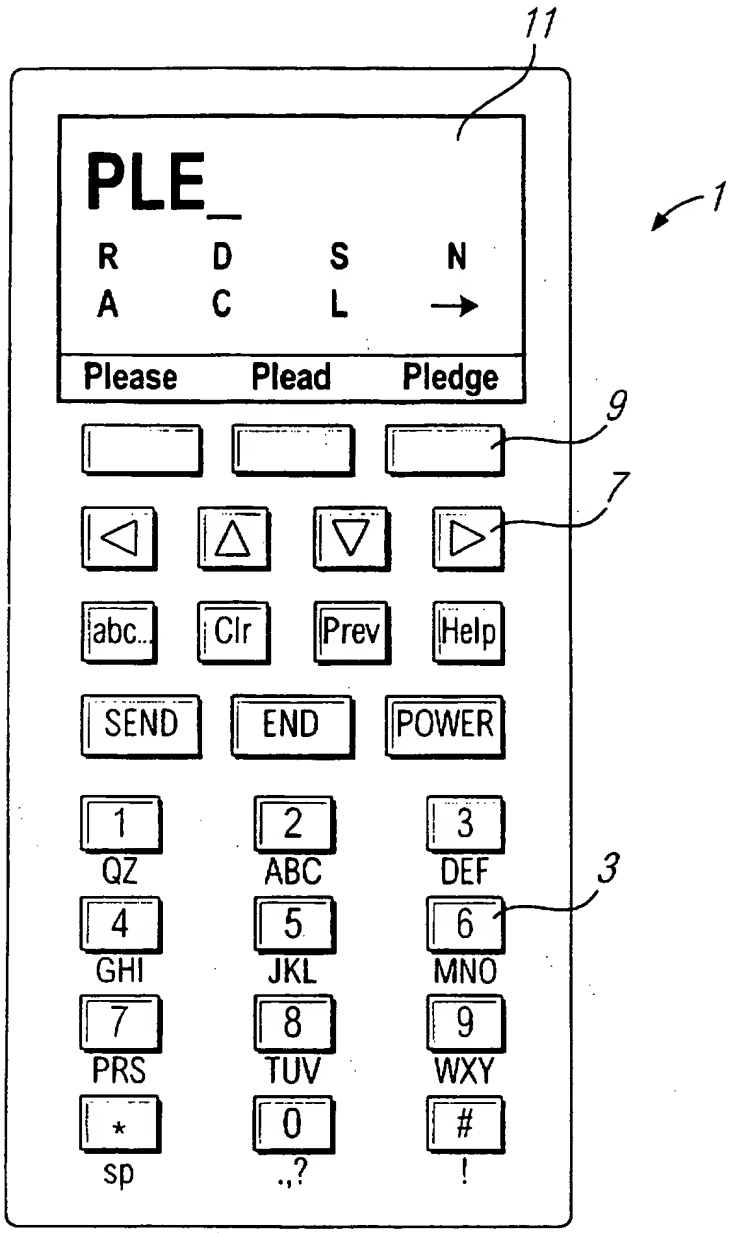

11
PLE_
R D S N
A C L →
Please Plead Pledge
1
9
7
abc Clr Prev Help
SEND END POWER
1 QZ
2 ABC
3 DEF
4 GHI
5 JKL
6 MNO
7 PRS
8 TUV
9 WXY
* sp
0 .,?
!
3

FIG. 4

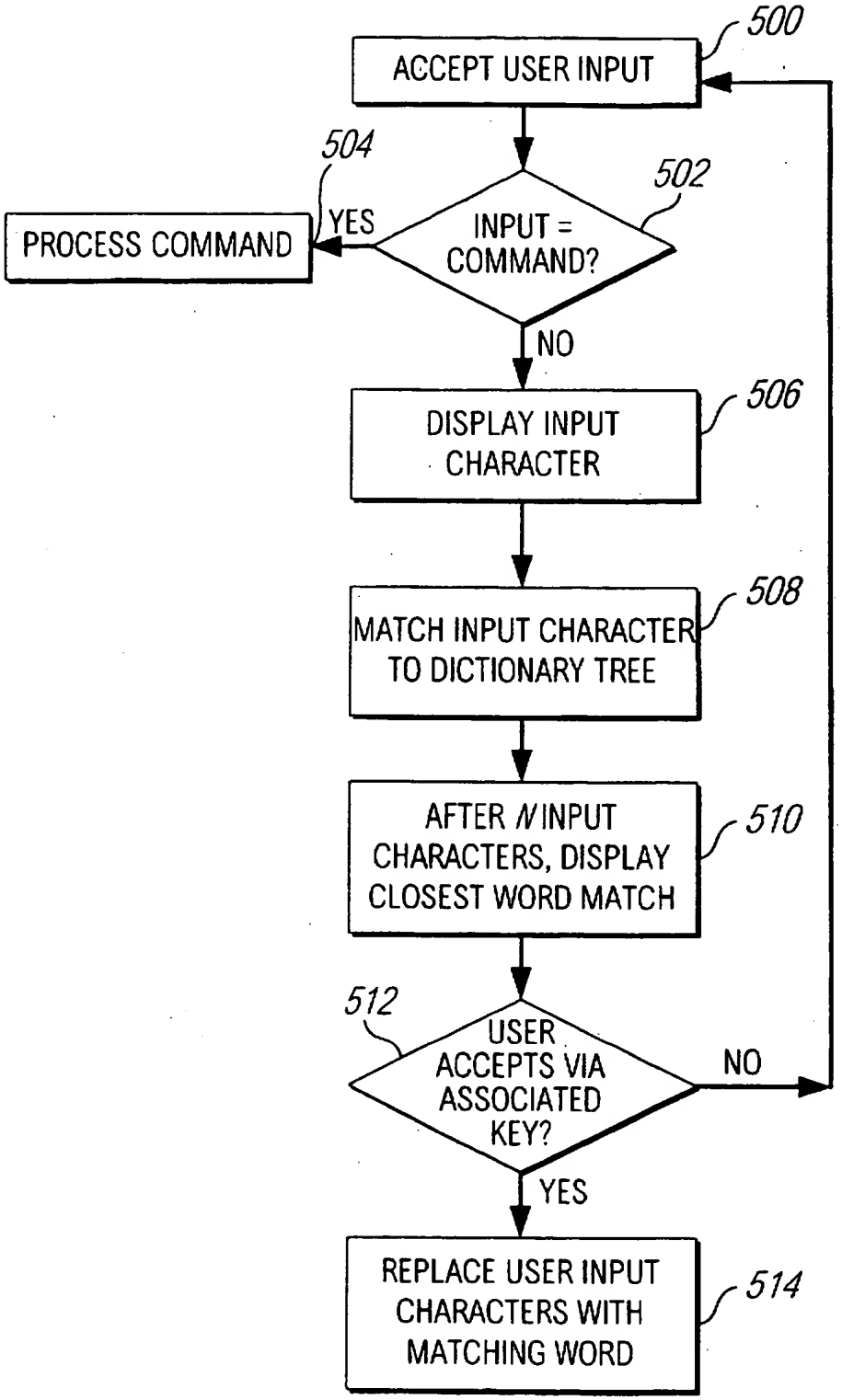

500
ACCEPT USER INPUT
504
502
YES
PROCESS COMMAND
INPUT = COMMAND?
NO
506
DISPLAY INPUT CHARACTER
508
MATCH INPUT CHARACTER TO DICTIONARY TREE
510
AFTER N INPUT CHARACTERS, DISPLAY CLOSEST WORD MATCH
512
USER ACCEPTS VIA ASSOCIATED KEY?
NO
YES
514
REPLACE USER INPUT CHARACTERS WITH MATCHING WORD

FIG. 5

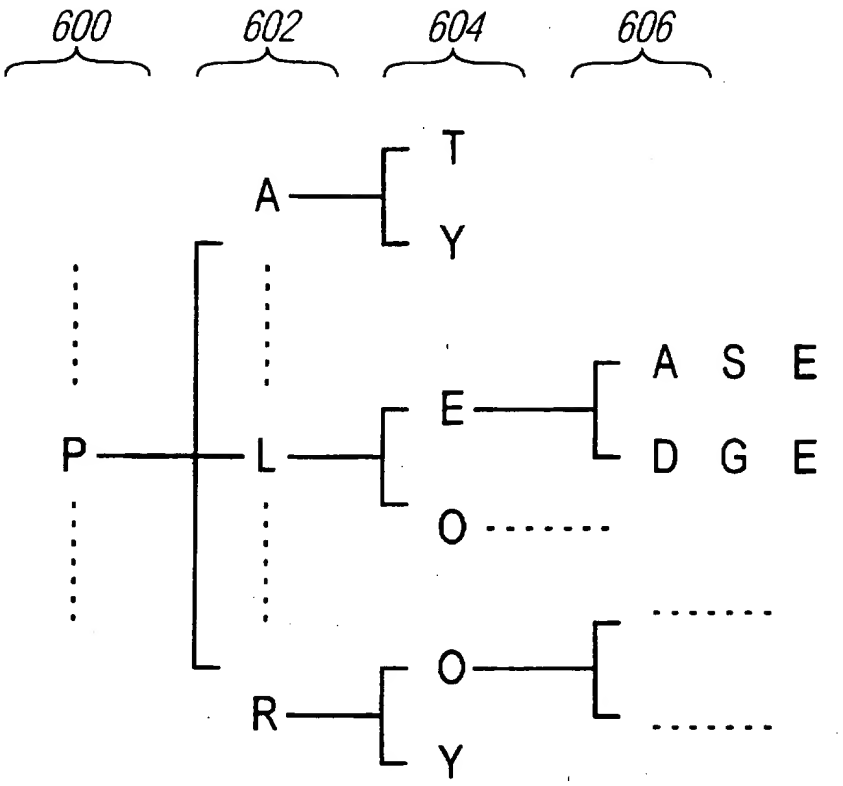


FIG. 6

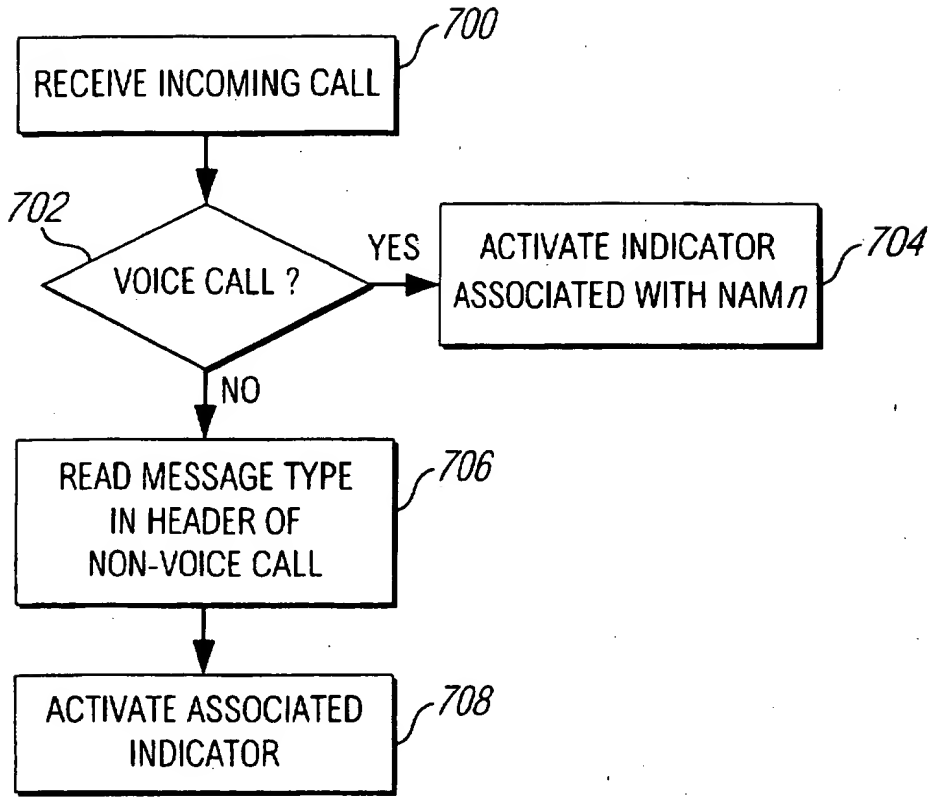

700
RECEIVE INCOMING CALL
702
VOICE CALL ?
YES
ACTIVATE INDICATOR ASSOCIATED WITH NAM n
704
NO
READ MESSAGE TYPE IN HEADER OF NON-VOICE CALL
706
ACTIVATE ASSOCIATED INDICATOR
708

FIG. 7

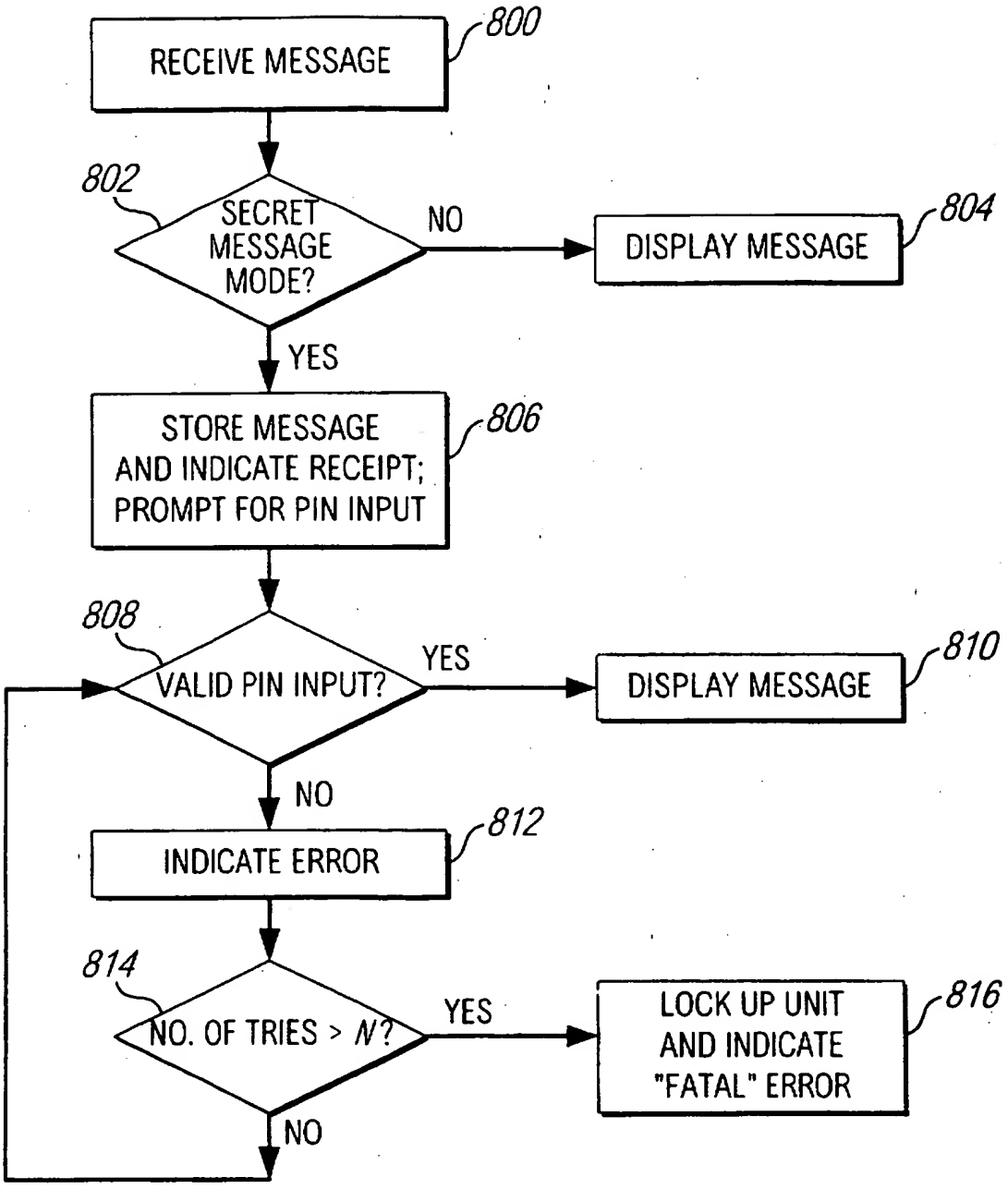

RECEIVE MESSAGE
800
802
SECRET MESSAGE MODE?
NO
DISPLAY MESSAGE
804
YES
STORE MESSAGE AND INDICATE RECEIPT; PROMPT FOR PIN INPUT
806
808
VALID PIN INPUT?
YES
DISPLAY MESSAGE
810
NO
INDICATE ERROR
812
814
NO. OF TRIES > N?
YES
LOCK UP UNIT AND INDICATE "FATAL" ERROR
816
NO

FIG. 8

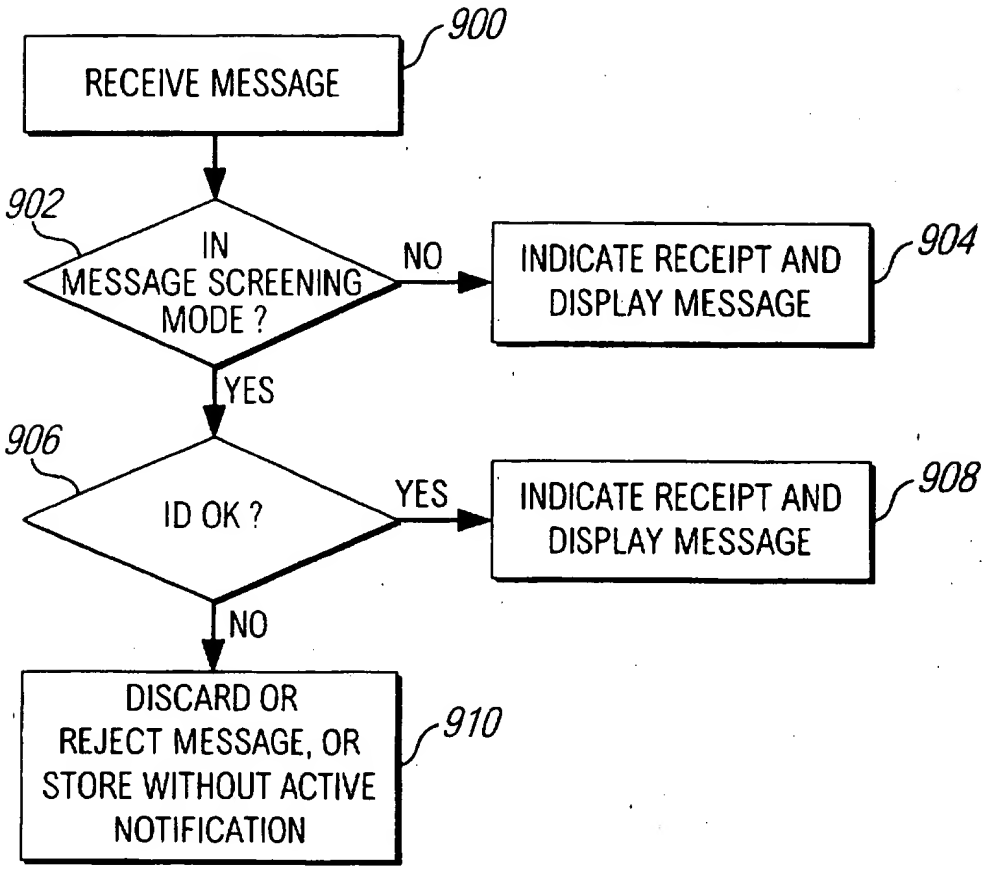

900
RECEIVE MESSAGE
902
IN MESSAGE SCREENING MODE ?
NO
INDICATE RECEIPT AND DISPLAY MESSAGE
904
YES
906
ID OK ?
YES
INDICATE RECEIPT AND DISPLAY MESSAGE
908
NO
DISCARD OR REJECT MESSAGE, OR STORE WITHOUT ACTIVE NOTIFICATION
910

FIG. 9

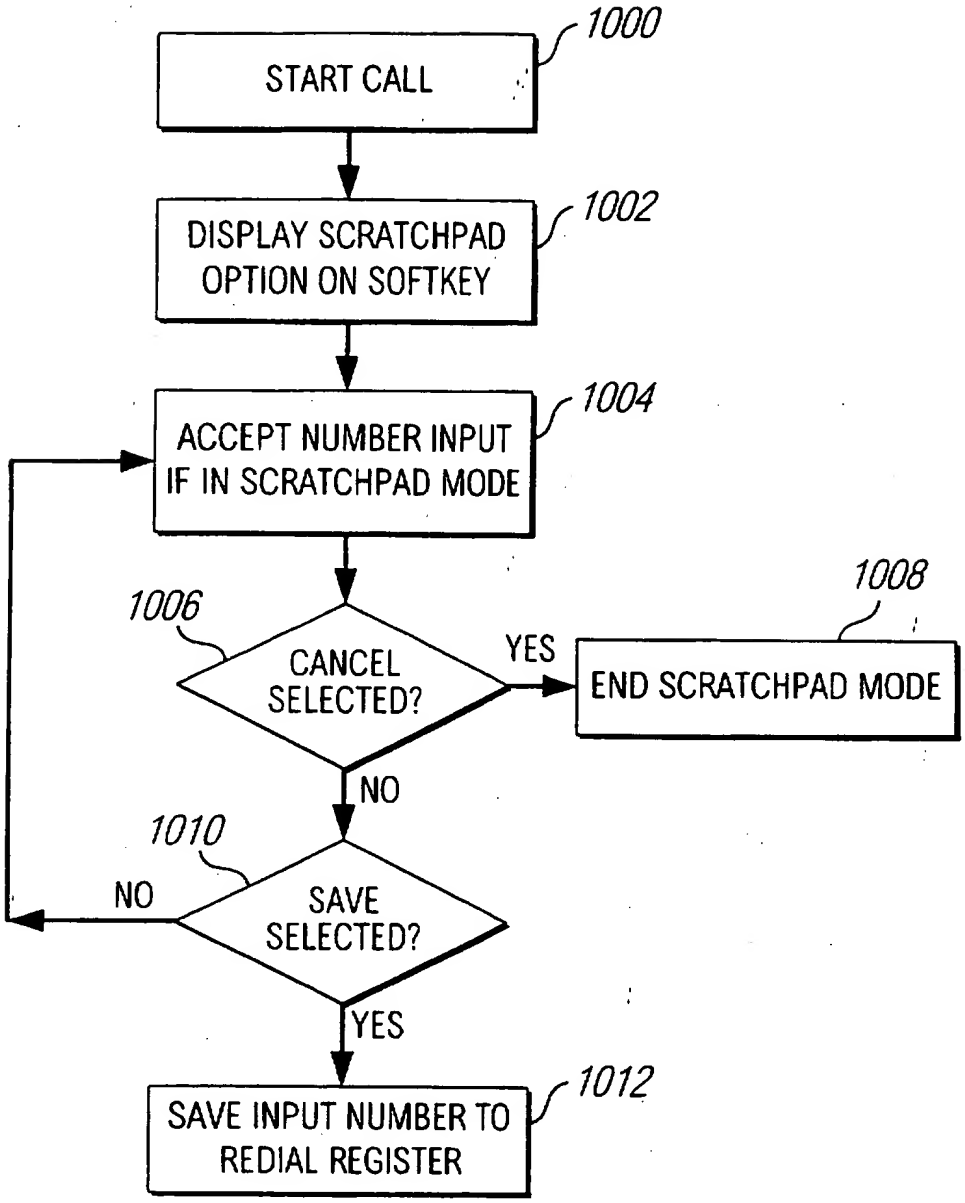

1000
START CALL
1002
DISPLAY SCRATCHPAD OPTION ON SOFTKEY
1004
ACCEPT NUMBER INPUT IF IN SCRATCHPAD MODE
1006
CANCEL SELECTED?
YES
1008
END SCRATCHPAD MODE
NO
1010
SAVE SELECTED?
NO
YES
1012
SAVE INPUT NUMBER TO REDIAL REGISTER

FIG. 10

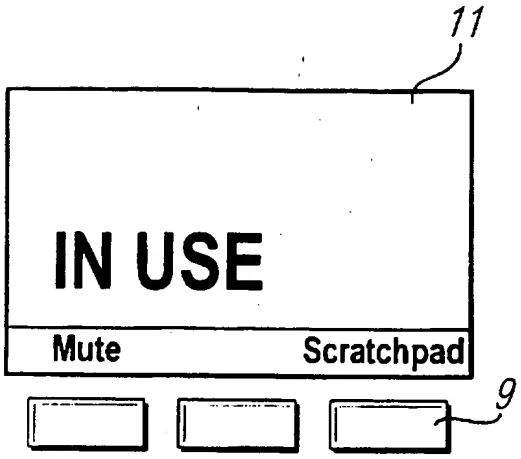


FIG. 11A

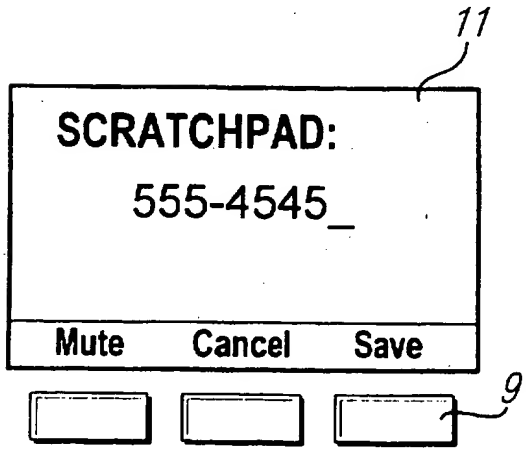


FIG. 11B

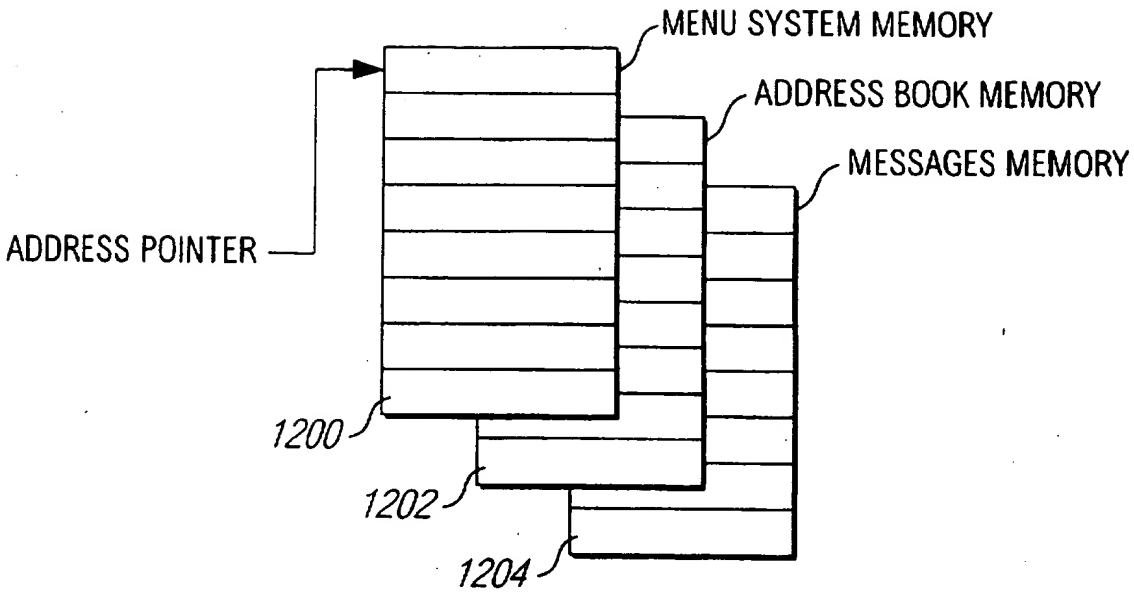

MENU SYSTEM MEMORY
ADDRESS BOOK MEMORY
MESSAGES MEMORY
ADDRESS POINTER
1200
1202
1204

FIG. 12

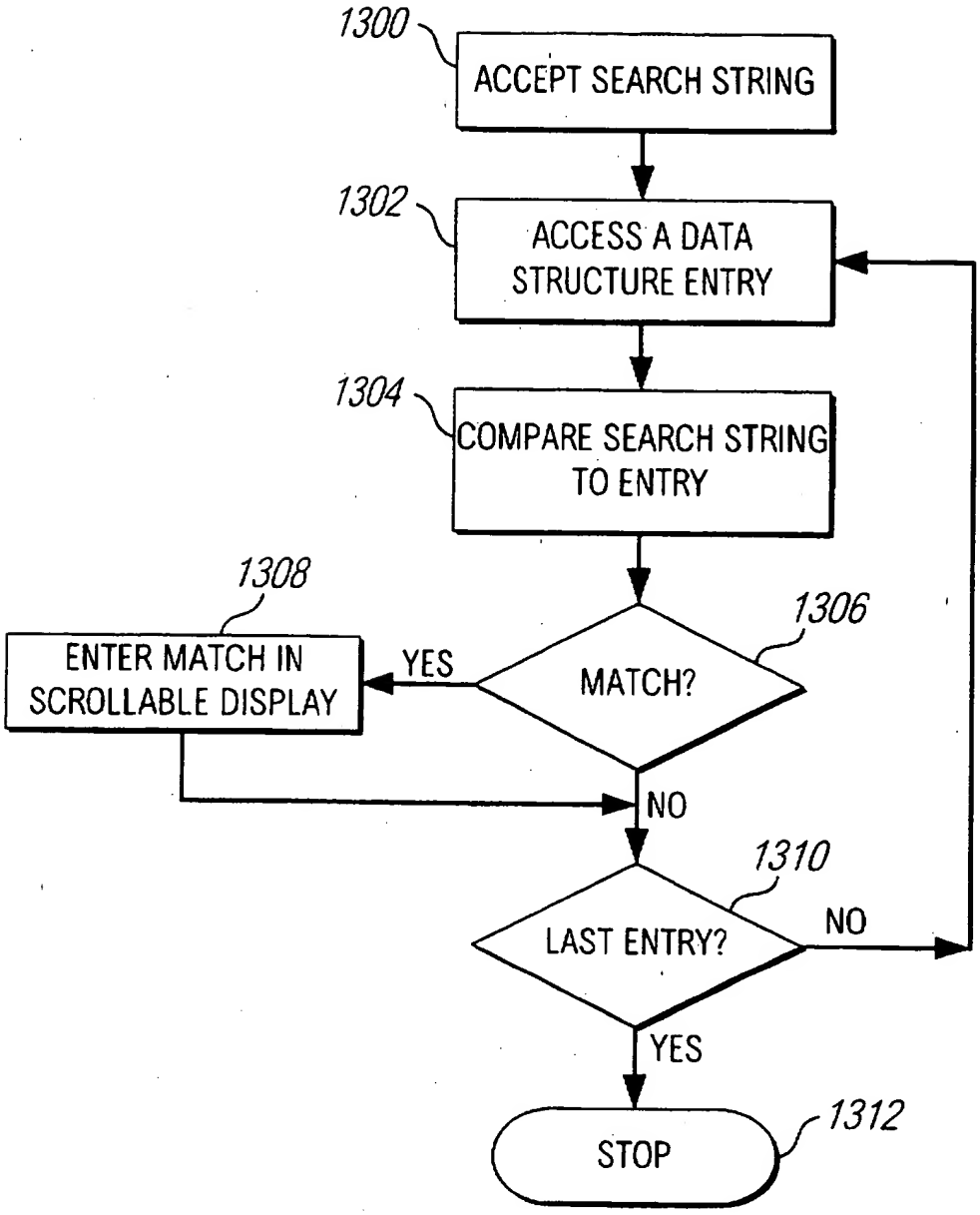

1300
ACCEPT SEARCH STRING
1302
ACCESS A DATA STRUCTURE ENTRY
1304
COMPARE SEARCH STRING TO ENTRY
1306
MATCH?
YES
1308
ENTER MATCH IN SCROLLABLE DISPLAY
NO
1310
LAST ENTRY?
NO
YES
1312
STOP

FIG. 13

USER INTERFACE FOR CELLULAR TELEPHONE

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates to electronic cellular telephone systems, and more particularly to an improved user interface for a cellular telephone system subscriber unit.

2. Description of Related Art

Cellular telephones have become very popular as communication devices. Typical cellular telephones are small, hand-held devices that provide a keypad for data entry and a display to see the results of such entry. As the popularity of hand-held cellular telephones has increased, the number of features offered by such devices has increased.

A problem with the numerous features offered by cellular telephones is that they are frequently difficult to use. Because of the limited keyboard (typically a 0–9 numeric keypad plus several additional function keys) and the limited amount of display (commonly only a single line), accessing and using such features as telephone directories, "canned" messages, security features, and the like are often confusing and difficult.

Accordingly, it is desirable to have a hand-held cellular telephone system that offers various "user-friendly" features that are easy to use despite the space limitations of a keyboard and display inherent in a hand-held design. The present invention overcomes these limitations by providing innovative ways in which to enter data and access a number of user convenient features, including improved search capabilities, security features, and alphanumeric data entry features.

SUMMARY OF THE INVENTION

An improved user interface for cellular or mobile phones, including the following functions:

(1) A predictive keyboard input method that initially displays to a user the most commonly used characters for selected words in a particular language (e.g., English), associating those characters with programmable keys, and then accepting the user's input. Thereafter, a next set of characters is displayed that are most likely to occur after the character previously input by the user.

(2) A word completion method in which a telephone displays user-defined or predefined candidate words matching initial characters input by a user, and assigns those candidate words to one or more available programmable keys. If one of the candidate words is the word being entered by the user, then the user need only press the associated programmable key to have the entirety of the candidate word automatically entered into the display.

(3) A distinctive signaling method useful in a dual-mode or tri-mode cellular phone system that incorporates both voice call functionality and data messaging functionality. Distinctive signaling is used to distinguish incoming voice calls from incoming data messages, and further to distinguish between various different data message types by examining header information in a received data message.

(4) A secret message method that permits secret messages to be received by a user of a cellular telephone that includes a data messaging capability. An incoming messages is received and stored before being displayed. All or selected received and stored messages are blocked from being displayed to persons other than the intended recipient by requiring input of a personal identification number or code.

In an alternative embodiment, a received message indicates its source and requires the receiver to input a code corresponding to the source, thus allowing each sender to "encrypt" a message with a key unique to the sender, which the receiving party must know in advance to "decrypt".

(5) A message screening method that permits a user to set a message screening mode in a cellular telephone. This mode allows screening of incoming data messages, such as in a CDPD system. When an incoming data message is received by a cellular telephone, an ID field from a header in the incoming data packet is examined to determine whether the source of the message is not from an "included" list or is from an "excluded" list. Screened out messages are discarded, flagged as low priority, or rejected before acceptance to lower user communication costs.

(6) An improved "scratchpad" method which permits a user to enter a telephone number into a storage register of a cellular telephone while in the middle of a voice call, visually verify the entry, and then save the number to a rapid redial location for later use.

(7) A global search method for searching text strings in all of the different memory sections of a cellular telephone having an address book, a hierarchical menu structure, and stored data messages.

The details of the preferred embodiment of the present invention are set forth in the accompanying drawings and the description below. Once the details of the invention are known, numerous additional innovations and changes will become obvious to one skilled in the art.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1*a* shows a front view of the preferred embodiment of a keyboard and display of a hand-held cellular telephone in accordance with the present invention.

FIG. 1*b* shows a block diagram of a processing system for implementing a cellular telephone of the type shown in FIG. 1*a*.

FIG. 2 shows a flowchart depicting a preferred method for implementing the predictive keyboard input process of the present invention.

FIG. 3 shows an example of use of the predictive keyboard input process of the present invention.

FIG. 4 shows a front view of the preferred embodiment of a keyboard and display of a hand-held cellular telephone in accordance with the present invention, showing an example of the word completion process of the present invention.

FIG. 5 shows a flowchart depicting a preferred method for implementing the word completion process of the present invention.

FIG. 6 shows a diagram of a portion of a dictionary tree, showing how the search function used for the word completion process of the present invention can be implemented.

FIG. 7 shows a flowchart depicting a preferred method for implementing the distinctive signaling process of the present invention.

FIG. 8 shows a flowchart depicting a preferred implementation of the secret message process of the present invention.

FIG. 9 shows a flowchart depicting a preferred implementation of the message screening process of the present invention.

FIG. 10 shows a flowchart depicting a preferred implementation of the improved scratchpad process of the present invention.

FIG. 11*a* is a front view of a cellular telephone showing a first example of use of the scratchpad process of the present invention.

FIG. 11*b* is a front view of a cellular telephone showing a second example of use of the scratchpad process of the present invention.

FIG. 12 is a diagram of three physical memory data structures used in accordance with the present invention.

FIG. 13 is a flowchart depicting a preferred implementation of the global search process of the present invention.

Like reference numbers and designations in the various drawings refer to like elements.

DETAILED DESCRIPTION OF THE INVENTION

Throughout this description, the preferred embodiment and examples shown should be considered as exemplars, rather than as limitations on the present invention.

FIG. 1*a* shows a preferred embodiment of the keyboard and display of a hand-held cellular telephone 1 (also known as a subscriber unit) in accordance with the present invention. A conventional telephone keypad 3 is provided, along with dedicated keys 5 for the functions "SEND", "END", and "POWER". In the preferred embodiment, the telephone 1 also includes a set of eight programmable keys 7 that have an initial dedicated function, but are also programmable in the sense that additional functions can be assigned to any of the eight keys. In addition, three "soft" keys 9 are provided. Text messages, icons, or the like may be associated with any one of the three soft keys 9 by displaying the associated symbol or text directly above a soft key 9 on a display 11, in known fashion.

The display 11 in the preferred embodiment is a multi-line (for example, 4 to 5 text lines) alphanumeric display that may be implemented as a dot matrix addressable display or a combination of a dot matrix addressable display and dedicated icons or symbols. In the preferred embodiment, the display 11 is implemented using a liquid crystal display, but other display technologies, such as LED, EL, etc., may be used.

The keys of the cellular telephone 1 may be implemented using any convenient keyboard technology, such as touch panel, membrane, mechanical, or optical switches.

FIG. 1*b* shows a block diagram of a programmable processing system 13 for implementing a cellular telephone 1 of the type shown in FIG. 1*a*. The processing system 13 preferably includes a CPU 20, a RAM 21, a ROM 22 (preferably writable, such as a flash ROM) and an I/O controller 23 coupled by a CPU bus. The I/O controller 23 is coupled by means of an I/O bus to a transmit/receive module 24. The transmit/receive module 24 is for receiving and transmitting analog carrier wave forms with analog or digital information imbedded therein, and decoding digital portions of such wave forms for output to the I/O bus, and analog portions of those wave forms for output to a speaker 26. Input to the transmit/receive module includes analog wave forms from a microphone 25. Also coupled to the I/O bus is a display 27 and a keyboard 28. The programmable processing system 13 may be pre-programmed, or may be programmed (and reprogrammed) by downloading a program from another source (e.g., another computer).

In the preferred embodiment, the cellular telephone 1 combines both conventional voice functionality (e.g., in accordance with AMPS - American Mobile Phone System - analog cellular receive and transmit functions) with digital data message capability (e.g., in accordance with the CDPD standard). In an alternative embodiment, both an analog voice cellular system, such as AMPS, is combined with a digital voice transmission system, using for example the TDMA (Time Division Multiple Access) or CDMA (Code Division Multiple Access) protocols, along with a data transmission system, such as CDPD. An example of such a tri-mode cellular telephone 1 is disclosed in co-pending U.S. patent application Ser. No. 08/117,913, assigned to the assignee of the present invention.

As FIG. 1*a* should make clear, both the keyboard and the display 11 have limited ability to enter or display data. Indeed, as is common, the telephone number keypad 3 has 2 or 3 letters or symbols assigned to virtually every key. Accordingly, the present invention addresses a variety of methods for overcoming the limitations of such a hand-held unit.

Predictive Keyboard Input

One problem specifically addressed by the present invention is the entry of text information into the display 11. For example, a directory for telephone numbers is provided in the preferred embodiment that permits associating a text string, such as the name of a person, with numeric data. Accordingly, some means must exist to enter such text. Furthermore, since the preferred embodiment of the present invention can be used for transmitting data according to a data transmission protocol, such as CDPD, it is important to be able to enter text messages for transmission.

In order to ease the problem of entering text using a limited keyboard, such as that shown in FIG. 1*a*, the present invention provides for predictive keyboard input. Predictive keyboard input provides a method for initially displaying to a user the most commonly used characters for selected words in a particular language (e.g., English), associating those characters with programmable keys, and then accepting the user's input. Thereafter, a next set of characters is displayed that are most likely to occur after the character previously input by the user.

Referring to FIG. 1*a*, a hand-held cellular telephone 1 is shown with a set of seven characters depicted in the display 11. The characters are paired and stacked vertically so as to correspond to seven of the eight programmable keys 7 on the telephone 1. The eighth display position is for a symbol or entry indicating "more", allowing a user to page to a different set of seven characters if a desired character is not in the group of seven currently displayed. An arrow symbol is shown in FIG. 1 to indicate the "more" function. The "more" symbol itself is assigned to one of the programmable keys 7. Thus, referring to FIG. 1, to select the character "A" for text entry, a user would push the programmable key 7 that physically corresponds to the displayed "A", being the "right arrow" cursor key in the top right-hand corner of the eight programmable keys 7. This physical mapping of the characters or symbols in the display 11 with the programmable keys 7 provides for a more intuitive interface for a user. In particular, each programmable key 7 is assigned one and only one character or symbol at any one time, in contrast to a conventional telephone keypad, which has 2 to 3 characters assigned to each button 3.

FIG. 2 shows a flowchart depicting a preferred method for implementing the predictive keyboard input process of the present invention. When in text entry mode, an initial character subset is displayed (STEP 200). The telephone 1 accepts user input from the keyboard (STEP 202). In the preferred embodiment, if the input is a command (such as

"SEND MESSAGE") (STEP 204), then the telephone 1 processes the command (STEP 206). If the input corresponds to the "more" function (STEP 208), then the display is changed to show the next most likely character subset (STEP 210), and processing continues back at STEP 202.

If the input is a keypress of a programmable key 7 corresponding to one of the displayed characters, that character is displayed on the display 11 as text entry (STEP 212). The display 11 is then changed to show the next most likely character subset, based on the previous character entry (STEP 210), and processing continues back at STEP 202. In the preferred embodiment, an optional word completion function (described below) may be implemented (STEP 214).

The determination of which characters to display in a particular subset can be made by statistical analysis of sample text from a particular language. For example, applicants have determined, from a set of particular English words commonly used for simple messages (e.g., "call", "meet", "today", "tomorrow", etc.), that the seven most common initial characters are those shown in the display 11 of FIG. 1. Pressing the "more" function key would display the next seven most common initial characters, and so forth. In the preferred embodiment, a table is constructed in memory having (1) the characters of the English alphabet ranked in order of their occurrence as initial characters of English text, and (2) an entry for every individual character of the alphabet, with each entry comprising a list of the most common characters occurring after each such character. For example, for the character "A", the list entry would be the 25 characters of the alphabet in order of their statistical occurrence after the character "A". The 25 characters would then be displayed 7 at a time in the illustrated embodiment of the invention. In an alternative embodiment of the present invention, the table can take into account the preceding N characters, rather than just the preceding single character entered by the user. For example, a table could be constructed whereby, if a user enters the characters "th", the frequency of characters occurring after that combination are determined and made entries in the table. Other rules for generating the character subsets may be adopted as desired.

In an alternative embodiment, the cellular telephone 1 allows a user to enter a list of words that the user frequently uses in messages (e.g., names, locations, products, etc.) and the internal CPU 20 builds a character frequency table from that specific set of words. In yet another embodiment, the cellular telephone 1 comes with a pre-defined set of character frequencies,which may be modified by analyzing the character frequencies of messages entered by a user over time) so that the table of character frequencies automatically adapts to the types of words used by the user.

FIG. 3 shows an example of the predictive keyboard input aspect of the present invention. Each right-hand entry in FIG. 3 represents characters that would be displayed in the display 11 of a hand-held unit in accordance with the present invention. The left-most column indicates either the initial display (row a), or the characters or symbols entered by a user (rows b–r). The example phrase to be entered is "PLEASE CALL HOME". As can be seen, the initial character "P" is one of the characters initially shown in the display 11. When the user presses the programmable key 7 corresponding to the character "P", the character subset shown in the display shifts to a new set, comprising the seven characters most likely to occur after the character "P". As can be seen in row b, the next character to be entered, "L", is already shown in the shifted display. Continuing on in this fashion, the entire phrase "PLEASE CALL" can be entered simply by selecting one of the programmable keys 7 corresponding to one of the characters shown in the display 11. Note that it is not until row n is reached that the "more" function has to be invoked, since the character "H" does not appear in the display after the "space" character is entered in row m. However, the character "H" does occur in the next displayed character subset after invoking the "more" function, as shown in row n. Thereafter, the rest of the message can be entered without having to use the "more" function.

Thus, the predictive keyboard method of the present invention allows a user of a hand-held cellular telephone 1 to easily enter text into the display 11 using programmable keys 7 that correspond to single characters or symbols at any one moment in time.

Word Completion

Although the predictive keyboard input method discussed above substantially reduces the difficulty of entering text into the limited keyboard and display of a cellular telephone 1, it still requires that the user enter each character of a word. Since cellular telephones are frequently used "on the go", it would be useful to have a method of speeding up text input. FIG. 4 shows a preferred embodiment of one such method, in which the telephone 1 displays candidate words matching initial characters input by a user, and assigns those candidate words to one or more available soft keys 9 and/or programmable keys 7 of the telephone 1. For example, as shown in FIG. 4, if the user has entered the letters "PLE" in the display 11 (either using prior art techniques, or the predictive keyboard input technique described above), a dictionary lookup routine may display the word "PLEASE", "PLEAD", and "PLEDGE" on a portion of the display 1 1 so that the candidate words are placed adjacent to associated soft keys 9. If one of the candidate words is the word being entered by the user, then the user need only press the associated soft key 9 to have the entirety of the candidate word automatically entered into the display 11. Thus, for example, if the user was entering the word "PLEASE", and had already entered the letters "PLE", the user need only press the left-most soft key 9 shown in FIG. 4 to enter the entirety of the word "PLEASE". As a convenience, a space character can be automatically placed after each completed word, and the first word after sentence-ending punctuation (e.g., period, question mark, exclamation mark, etc.) can be displayed with an initial capital letter.

FIG. 5 shows a flowchart of this process. The telephone 1 accepts the user's input (STEP 500). If the input is a command (STEP 502), then the command is processed in conventional fashion (STEP 504). Otherwise, the input is displayed as a character (STEP 506). The processing system 13 within the unit then matches the input character to a dictionary tree (described below) (STEP 508). As soon as enough characters N (for example, 2 or 3 characters) have been input by the user to determine at least one plausible candidate word in the dictionary tree, one or more of the candidate words are associated with corresponding soft keys 9 and shown in the display 11 of the telephone 1 (STEP 510). In the example shown in FIG. 4, up to three candidate words may be displayed in the bottom-most portion of the display 11 such that the candidate words are associated with the three soft keys 9. If the available display area is larger, then additional candidate words can be shown. For example, if room for two rows of candidate words were available, the programmable keys 7 could be used instead of the soft keys 9 to be associated with or mapped to the candidate words.

Thereafter, if the user accepts one of the candidate words by pressing an associated soft key 9, the current word being

input by the user is replaced by the candidate word, and displayed in the display 11 (STEP 514). The user can continue entering text or commands as desired. If the user does not accept one of the candidate words (STEP 512), additional input is accepted from the user (STEP 500).

In an alternative embodiment, if no match occurs after a preset number of characters has been entered, the list of candidate words shown in display 11 is cleared to avoid confusion.

FIG. 6 shows a diagram of a portion of a dictionary tree, showing how the search function used for the word completion can be implemented. In the example shown, a first tier 600 has individual letters of the alphabet, the letter "P" being shown. A next tier 602 of the dictionary tree contains letters of words in the dictionary that follow the letter of the preceding tier. In the example shown, the letters "A", "L", and "R" are shown. Additional tiers 604, 606 may be used to extend the dictionary tree, as shown in FIG. 6. The "leaf" nodes of the dictionary tree may contain single letters (such as the entry "P-A-T") or multiple letters (such as the entry "P-L-E-ASE"). The dictionary tree may be as extensive as desired. Although a dictionary tree is shown, other methods may be used for matching input characters to plausible candidate words, such as a dictionary look-up table. The implementation of such dictionary trees or lookup tables in the computer arts is well known.

In the preferred embodiment, the dictionary tree is predefined, but modifiable by the user. That is, the user can enter words (e.g., names, locations, products, etc.) commonly used by the user, and the CPU 20 is programmed to index such words into the dictionary tree in known fashion. Thus, if the user frequently inputs the word "Massachusetts", the user places the telephone 1 into an edit mode, enters the characters for "Massachusetts", and the CPU 20 inserts a leaf node and pointers into the dictionary tree so that "Massachusetts" is after "major" and before "master" (as an example only).

Alternatively, the words in the dictionary can be statistically determined in part from sample text, including text previously input by the user into the cellular telephone unit. That is, new words input by the user can be automatically added to the dictionary based on frequency of input of such new words. For example, if the telephone 1 can hold 100 new words, all new words not found in the pre-defined dictionary would be automatically indexed and added to the dictionary. In addition, the frequency of use of each such word can be counted by associating an additional count or time stamp field with each word. Once the new word capacity of the telephone 1 is exhausted, a least recently used algorithm can be used to replace infrequently used words with new words more recently input by the user.

Thus, the word completion aspect of the present invention provides a means for rapid entry of common words into a hand-held cellular telephone 1 without requiring keying each letter of such words.

Distinctive Signaling

Another aspect of the present invention useful in a dual-mode or tri-mode cellular phone system that incorporates both voice call functionality and data messaging functionality is an improved form of distinctive signaling. In prior art land line telephone systems, distinctive signaling provides a unique auditory signal on one telephone to indicate incoming calls from one of several input lines. In the present context, distinctive signaling is used to distinguish incoming voice calls from incoming data messages, and further to distinguish between various different possible types of data messages.

A flowchart of the inventive distinctive signaling method is shown in FIG. 7. The cellular telephone 1 receives an incoming call in conventional fashion (STEP 700). If the call is a conventional analog voice call (e.g., from an AMPS system), the telephone 1 will use conventional analog voice circuitry to receive that call. Since only the voice circuitry is activated, it can be determined that the incoming call is voice and not data (STEP 702). Further, it can be determined whether an incoming voice call is associated with a particular NAM ("Number Assignment Module"). This is done in known fashion by monitoring the analog voice system control channel. Consequently, a different indicator can be associated with each NAM available within a cellular telephone unit 1 (STEP 704). The indicator may be, for example, auditory (such as conventional distinctive ringing), vibratory (to give a tactile indication to a user without being conspicuous), visual (such as different blinking patterns or messages in the display 11), or any combination of the above, such that the indicator associated with a particular NAM is distinctive.

If the incoming call is not a voice call, but is instead a data call, the header of the data packet (such as the header of a conventional CDPD data packet) is examined to determine the message type (STEP 706). For example, the preliminary (5-10-95) Protocol Specification for a Short Message Service Over CDPD ("SMS-CDPD"), published by the assignee of the present invention and herein incorporated by reference, specifies one header structure that can be used by the present invention. As one example, only, the SMS-CDPD protocol specifies an application layer header having a code that indicates that an incoming message is a pager call, a voice mail notification call, or a data message call. Other information that may be stored in an incoming data packet may include the address, source, and/or class of the message. Accordingly, since each message can be distinguished by type and/or by source, a distinctive indication can be associated with each such message type to alert the user not only that a data message call has been received, but the nature of the message (STEP 708). Again, as described above, the indicator associated with a particular message type may be auditory, vibratory, visual, or a combination of the above.

Secret Message Processing

Another aspect of the present invention permits secret messages to be received by a user. This aspect of the invention is used in conjunction with cellular telephones that include a data messaging capability. In such systems, incoming messages can be received and stored before being displayed. The present invention takes advantage of this characteristic by providing a way for preventing all or selected received and stored messages from being displayed to persons other than the intended recipient.

FIG. 8 shows a flowchart showing the preferred implementation of the secret message process of the present invention. A cellular phone system of the type shown in FIG. 1*a* initially receives a data message (STEP 800). The telephone 1 can test to see whether or not it has been set to a secret message mode (STEP 802), if all incoming messages are to be held secret until authorized for display. Alternatively, the telephone 1 may be programmed to inspect a header of each incoming data packet to determine if a "secret" or "private" secrecy flag or code has been set by the sender.

If a received message is not flagged as secret or private, or the telephone 1 is not set to a general secret message

mode, the message is displayed in conventional fashion (STEP 804). If a received message is flagged as secret or private, or if the secret message mode has previously been activated by the user, the message is stored and an indication of receipt given to the user. The telephone 1 also prompts the user for an authorization code, such as a Personal Identification Number ("PIN") (STEP 806). If the user inputs a valid PIN (STEP 808), the stored message is displayed (STEP 810). If a valid PIN has not been input, an error is indicated (STEP 812). As an optional security measure, if the number of attempts to input the PIN exceeds a preset number N (STEP 814), the cellular telephone 1 disables itself and indicates a "fatal" error requiring dealer service (STEP 816). Otherwise, the user is re-prompted for a valid PIN input.

In an alternative embodiment, a received secret message indicates its source (e.g., by name, telephone number, or some other code) and requires the receiver to input a code corresponding to the source. This allows each sender to "encrypt" a message with a key unique to the sender, which the receiving party must know in advance to "decrypt". For example, a secret message sent by a remote office manager to the recipient can require that the recipient unlock the message by inputting the office manager's coder rather than the recipient's PIN.

Accordingly, this aspect of the present invention provides security for incoming data messages not provided by the prior art.

Message Screening Process

Another aspect of the present invention permits a user to set a message screening mode in a cellular telephone. This mode allows screening of incoming data messages, such as in a CDPD system. FIG. 9 shows a flowchart of the preferred implementation of this aspect of the invention. When an incoming data message is received by a cellular telephone 1 (STEP 900), the telephone 1 determines whether it is in the "message screening" mode (STEP 902). If not, the telephone 1 indicates receipt of the message and displays the message in conventional fashion (STEP 904).

If the telephone 1 is in message screening mode, an ID field from a header in the incoming data packet is examined to determine whether the source of the message is from a user-defined "included" list (STEP 906). In an alternative embodiment, the source ID of the incoming message is examined to determine whether it is from a user-defined "excluded" list.

If the received message is not to be screened out, then the message is displayed and the telephone 1 indicates receipt of the message to the user (STEP 908). If the received message is not from the included list (or alternatively is from the excluded list), then the message can be discarded by erasing it from the memory of the cellular telephone 1 (STEP 910). Alternatively, the message may be classified as "low priority", and saved for later perusal by the user, with only a nominal passive indication (e.g., visual only) being given that such a "low priority" message has been received. This prevents the user from being bothered by active notification (e.g., beeping) for unimportant incoming messages.

In the preferred embodiment of this aspect of the invention, only the header of the incoming data packets need be examined. If the message is to be rejected as being on an excluded list or not on an included list, as determined by examining the header information, a NAK (No Acknowledgment) or similar signal can be sent back immediately to the source transmitter, so that the user is not billed for receipt of an unwanted call.

The user can define included or excluded sources of messages by simply entering a code corresponding to the included or excluded source into a list in the memory of the cellular telephone 1. The source ID from incoming messages would then be compared to the entries in this list, in known fashion.

Improved Scratchpad

Another aspect of the present invention is an improved "scratchpad" feature which permits a user to enter a telephone number into a storage register of a cellular telephone while in the middle of a voice call. The entered number is then readily accessible for dialing. This feature is particularly useful if the user is talking to another person who gives the user another telephone number to dial. The user can immediately enter that telephone number into the telephone unit while continuing the voice call, such that the number is readily available to dial after termination of the current call.

FIG. 10 shows a flowchart describing the improved scratchpad process. The process starts by the user initiating a call (STEP 1000). The cellular telephone 1 preferably displays a "SCRATCHPAD" option in the display 11 so as to be associated with a soft key 9, as shown in FIG. 11*a* (STEP 1002). If the user selects the SCRATCHPAD option, the display 11 changes to indicate that the scratchpad feature has been activated, as shown in FIG. 11*b*. In this mode, normal dialing is disabled, so that numeric input by the user using the standard telephone keypad 3 on the telephone 1 is not interpreted as further numbers to dial or transmit. The display 11 can then show such numeric input from the user. The input telephone number is displayed on the display 11, as shown in FIG. 11*b*, for verification by the user (STEP 1004).

In the preferred embodiment, a "CANCEL" function is associated with one soft key 9, and a "SAVE" function is associated with another soft key 9, as shown in FIG. 11*b*. If the user selects the CANCEL function while in scratchpad mode (STEP 1006), then the scratchpad mode is terminated (STEP 1008). Otherwise, input continues until the user selects the SAVE function (STEP 1010). Once SAVE is selected, the input number is saved in an internal register within the cellular telephone unit, such as the "last redial" register (STEP 1012), and the display returns to the normal "in use" mode.

Thereafter, the user can have the cellular telephone 1 dial the number stored during the scratchpad process by invoking, for example, the "last redial" function, in conventional fashion. This may be done, for example, in the preferred embodiment, by pressing the "up" or "down" cursor function keys 7, shown in FIG. 1*a*. Alternatively, the system can be programmed such that pressing SEND after invoking the scratchpad process and recording a number will automatically dial that saved number.

This aspect of the present invention thus allows a user to record a desired telephone number during a call with positive visual feedback of the input number and easy access to the input number for dialing a next call.

Global Search

Another aspect of the present invention provides a means for searching text strings in the different memory sections of a cellular telephone 1 in accordance with the present invention. As the number of features in such telephones 1 proliferate, the amount of memory dedicated to data storage has increased significantly. A problem with the increase of such memory is that it is sometimes difficult to locate desired

data. This is particularly a problem where the memory may be partitioned into different blocks for different purposes. For example, in FIG. 12, the preferred embodiment of the present invention divides physical memory into at least three discrete sections, comprising Menu System Memory **1200**, Address Book Memory **1202**, and Messages Memory **1204**. The Menu System Memory **1200** contains text for the menu system command structure used for control of a cellular telephone **1** in accordance with the present invention. The Address Book Memory **1202** contains storage for data entries relating to conventional address information, such as name and telephone number. The Messages Memory **1204** provides storage for incoming and outgoing data messages for use in cellular phone systems that have a data messaging capability, such as under the CDPD standard.

As is known in the art, it is common to divide such memory into a data structure comprising words or blocks, accessible by an address pointer. The present invention provides a means for searching for text strings in the data structure of a plurality of memory sections of a cellular telephone **1**.

It is also commonly known in the art to divide data into logical structures. For example, the cellular telephone **1** may have a complex, multi-level or hierarchical menu structure. One problem of such is structure, especially with a limited display **11**, is remembering where particular menu functions are in the hierarchical structure. Accordingly, the present invention allows a user to directly search for a keyword that would appear on a menu or the text associated with a menu command, without having to traverse through the menu hierarchy itself

FIG. **13** is a flowchart showing the global search capability of the present invention. A user enters a desired search string, which is accepted into the cellular telephone **1** (STEP **1300**). The search algorithm may work at the physical storage level (such as is shown in FIG. **12**), or at a logical storage level (e.g., by traversing singly or doubly linked lists, sequential data records, hierarchical trees, etc.). In either approach, each addressable entry of the physical or logical data structure is accessed (STEP **1302**), and the contents of each entry are compared to the search string (STEP **1304**). The algorithm for comparing may be any desired algorithm known in the art. In the preferred embodiment, if a match occurs between the search string and an entry in one of the data structures (STEP **1306**), the entire contents of the data structure entry can be entered into a scrollable display so that the user can review the search results at will (STEP **1308**). Alternatively, the first match is displayed to the user, and one of the soft keys **9** or programmable keys **7** can be designated as a NEXT function, to search for the next match. If no match occurs, the search continues at STEP **1310**. If the last data structure entry has been searched (STEP **1310**), the process stops (STEP **1312**). Otherwise, the process continues at STEP **1302**.

In the preferred embodiment, a search is conducted concurrently through the data structures comprising the address book, hierarchical menu structure, and stored data messages of the telephone **1**. In an alternative embodiment, a search is conducted only through the data structure comprising the hierarchical menu structure of the telephone **1**. In yet another alternative embodiment, a search is conducted only through the data structure comprising the stored data messages of the telephone **1**.

This aspect of the present invention thus allows a user to readily search memory data structures within a cellular telephone regardless of whether the memories contain menu information, address book information, or message information. This unified, global search scheme means that the user can search for information in a plurality of memory structures without having to know in advance in which memory structure the information is likely to be.

Summary

The procedures described above are preferably implemented as computer programs stored on a storage media or device (e.g., ROM) readable by the programmable processing system **13**, for configuring and operating the programmable processing system **13** when the storage media or device is read by the programmable processing system **13**, the programmable processing system **13** being operated to perform the user interface functions described above. A number of embodiments of the present invention have been described. Nevertheless, it will be understood that various modifications may be made without departing from the spirit and scope of the invention. For example, a number of the features of the invention can be used with any hand-held communication device, such as two-way pagers, wireless personal digital assistants, cordless telephones, etc. Accordingly, it is to be understood that the invention is not to be limited by the specific illustrated embodiment, but only by the scope of the appended claims.

The invention claimed is:

1. In a cellular telephone having an alphanumeric display and a keyboard having a plurality of programmable keys, a predictive keyboard input method comprising the steps of:

(a) in a text entry mode, displaying an initial character subset, comprising a plurality of characters on the display;

(b) accepting an input keystroke from the keyboard;

(c) if the input keystroke corresponds to one of the displayed plurality of characters, then:

(1) displaying that character as an input character on the display; and

(2) displaying a next character subset, comprising a plurality of characters, on the display, wherein the characters in the next character subset are determined to be the next most probable characters based on at least one preceding input character;

(d) matching each input character to a dictionary of candidate words stored within the cellular telephone to select at least one candidate word in the dictionary;

(e) displaying at least one selected candidate word on the display, each selected candidate word being visually associated with a corresponding one of the plurality of programmable keys;

(f) accepting a next input keystroke from the keyboard;

(g) if such next input keystroke corresponds to one of the plurality of programmable keys, then displaying the selected candidate word associated with such programmable key as text entry on the display, replacing immediately preceding displayed input characters used to select the selected candidate word; and

(h) otherwise continuing at step (c), wherein the initial character subset is statistically determined from sample text to be the most common initial characters of words appearing in such sample text and the initial character subset is periodically updated by analyzing the character frequencies of messages entered by a user over time.

2. The invention of claim **1**, wherein the initial character subset is statistically determined from sample text to be the most common initial characters of words appearing in such sample text.

3. The invention of claim 1, wherein each next character subset is statistically determined from sample text.

4. The invention of claim 1, including the further step of automatically adding new words input by the user to the dictionary.

5. In a cellular telephone having an alphanumeric display and a keyboard having a plurality of programmable keys, a predictive keyboard input method comprising the steps of:

(a) in a text entry mode, displaying an initial character subset, comprising a plurality of characters, on the display, each character in the initial character subset being visually associated with a corresponding one of the plurality of programmable keys;

(b) accepting an input keystroke from the keyboard;

(c) if the input keystroke corresponds to one of the plurality of programmable keys, then:

(1) displaying the character associated with such programmable key as an input character on the display; and

(2) displaying a next character subset, comprising a plurality of characters, on the display, each character in the next character subset being visually associated with a corresponding one of the plurality of programmable keys, wherein the characters in the next character subset are determined to be the next most probable characters based on at least one preceding input character;

(d) matching each input character to a dictionary of candidate words stored within the cellular telephone to select at least one candidate word in the dictionary;

(e) displaying at least one selected candidate word on the display, each selected candidate word being visually associated with a corresponding one of the plurality of programmable keys;

(f) accepting a next input keystroke from the keyboard;

(g) if such next input keystroke corresponds to one of the plurality of programmable keys, then displaying the selected candidate word associated with such programmable key as text entry on the display, replacing immediately preceding displayed input characters used to select the selected candidate word; and

(h) otherwise continuing at step (c), wherein the initial character subset is statistically determined from sample text to be the most common initial character of words appearing in such sample text and the initial character subset is periodically updated by analyzing the character frequencies of messages entered by a user over time.

6. A cellular telephone having an alphanumeric display, a keyboard having a plurality of programmable keys, and a programmable processing system, the programmable processing system being programmed to:

(a) in a text entry mode, display an initial character subset, comprising a plurality of characters, on the display;

(b) accept an input keystroke from the keyboard;

(c) if the input keystroke corresponds to one of the displayed plurality of characters, then:

(1) display that character as an input character on the display; and

(2) display a next character subset, comprising a plurality of characters, on the display, wherein the characters in the next character subset are determined to be the next most probable characters based on at least one preceding input character;

(d) match the input character to a dictionary of candidate words stored within the cellular telephone to select at least one candidate word in the dictionary;

(e) display at least one selected candidate word in the display, each selected candidate word being visually associated with a corresponding one of the plurality of programmable keys;

(f) accept a next input keystroke from the keyboard;

(g) if such next input keystroke corresponds to one of the plurality of programmable keys, then display the selected candidate word associated with such programmable key as text entry on the display, replacing immediately preceding displayed input characters used to select the selected candidate word;

(h) otherwise continuing at step (c), wherein the initial character subset is statistically determined from sample text to be the most common initial characters of words appearing in such sample text and the initial character subset is periodically updated by analyzing the character frequencies of messages entered by a user over time.

7. The invention of claim 6, wherein each next character subset is statistically determined from sample text.

8. The invention of claim 6, wherein the programmable processing system performs the further function of automatically adding new words input by the user to the dictionary.

9. A cellular telephone having an alphanumeric display, a keyboard having a plurality of programmable keys, and a programmable processing system, the programmable processing system being programmed to:

(a) in a text entry mode, display an initial character subset, comprising a plurality of characters, on the display, each character in the initial character subset being visually associated with a corresponding one of the plurality of programmable keys;

(b) accept an input keystroke from the keyboard;

(c) if the input keystroke corresponds to one of the plurality of programmable keys, then:

(1) display the character associated with such programmable key as an input character on the display; and

(2) display a next character subset, comprising a plurality of characters, on the display, each character in the next character subset being visually associated with a corresponding one of the plurality of programmable keys, wherein the characters in the next character subset are determined to be the next most probable characters based on at least one preceding input character;

(d) match the input character to a dictionary of candidate words stored within the cellular telephone to select at least one candidate word in the dictionary;

(e) display at least one selected candidate word on the display, each selected candidate word being visually associated with a corresponding one of the plurality of programmable keys;

(f) accept a next input keystroke from the keyboard;

(g) if such next input keystroke corresponds to one of the plurality of programmable keys, then display the selected candidate word associated with such programmable key as text entry on the display, replacing mediately preceding displayed input characters used to select the selected candidate word;

(h) otherwise continuing at step (c), wherein the initial character subset is statistically determined from sample text to be the most common initial characters of words appearing in such sample text and the initial character subset is periodically updated by analyzing the character frequencies of messages entered by a user over time.

10. A computer program, for use in a cellular telephone having an alphanumeric display, a keyboard having a plurality of programmable keys, and a programmable processing system, for configuring and operating the programmable processing system to:

(a) in a text entry mode, display an initial character subset, comprising a plurality of characters, on the display;

(b) accept an input keystroke from the keyboard;

(c) if the input keystroke corresponds to one of the displayed plurality of characters, then:

(1) display that character as an input character on the display; and

(2) display a next character subset, comprising a plurality of characters, on the display, wherein the characters in the next character subset are determined to be the next most probable characters based on at least one preceding input character;

(d) match the input character to a dictionary of candidate words stored within the cellular telephone to select at least one candidate word in the dictionary;

(e) display at least one selected candidate word on the display, each selected candidate word being visually associated with a corresponding one of the plurality of programmable keys;

(f) accept a next input keystroke from the keyboard;

(g) if such next input keystroke corresponds to one of the plurality of programmable keys, then display the selected candidate word associated with such programmable key as text entry on the display, replacing immediately preceding displayed input characters used to select the selected candidate word;

(h) otherwise continuing at step (c), wherein the initial character subset is statistically determined from sample text to be the most common initial characters of words appearing in such sample text and the initial character subset is periodically updated by analyzing the character frequencies of messages entered by a user over time.

11. A computer program, for use in a cellular telephone having an alphanumeric display, a keyboard having a plurality of programmable keys, and a programmable processing system, for configuring and operating the programmable processing system to perform the functions of:

(a) in a text entry mode, display an initial character subset, comprising a plurality of characters, on the display, each character in the initial character subset being visually associated with a corresponding one of the plurality of programmable keys;

(b) accept an input keystroke from the keyboard;

(c) if the input keystroke corresponds to one of the plurality of programmable keys, then:

(1) display the character associated with such programmable key as an input character on the display; and

(2) display a next character subset, comprising a plurality of characters, on the display, each character in the next character subset being visually associated with a corresponding one of the plurality of programmable keys, wherein the characters in the next character subset are determined to be the next most probable characters based on at least one preceding input character;

(d) match the input character to a dictionary of candidate words stored within the cellular telephone to select at least one candidate word in the dictionary;

(e) display at least one selected candidate word on the display, each selected candidate word being visually associated with a corresponding one of the plurality of programmable keys;

(f) accept a next input keystroke from the keyboard;

(g) if such next input keystroke corresponds to one of the plurality of programmable keys, then display the selected candidate word associated with such programmable key as text entry on the display, replacing immediately preceding displayed input characters used to select the selected candidate word;

(h) otherwise continuing at step (c), wherein the initial character subset is statistically determined from sample text to be the most common initial characters of words appearing in such sample text and the initial character subset is periodically updated by analyzing the character frequencies of messages entered by a user over time.

* * * * *

EXHIBIT E

II. SPECIFICATION AMENDMENTS

Please amend the Abstract according to the following

ABSTRACT

A communication terminal having a display; a keypad having a plurality of keys associated with several letters each; a processor ~~means~~for controlling the display in accordance with the operation of the keypad; a selectable predictive editor program for generating an output containing words matching a received string of ambiguous key strokes, ~~said~~the predictive editor program has a number of associated vocabularies including at least one language dependent dictionary and at least one dictionary receiving user defined inputs. An editor application is controlled by the processor ~~means~~which communicates with said predictive editor programs for generating matching words based on an ambiguous string of key strokes. A ~~S~~second memory of the communication terminal for storing user inputted data. The processor ~~means~~ automatically searches the~~said~~ second memory ~~means~~ for words and copies these words into ~~said at least one~~the dictionary for receiving user defined inputs ~~and~~ associated with said predictive editor program.